

OPERA-3D USER GUIDE

Vector Fields Limited
24 Bankside
Kidlington
Oxford OX5 1JE
England

Copyright © 1999-2002 by Vector Fields Limited, England

This document was prepared using Adobe® FrameMaker®.

UNIX is a trademark of X/Open Company Ltd.

HPGL is a trademark of Hewlett-Packard Incorporated

PostScript is a trademark of Adobe Systems Incorporated

Windows 95/98/NT is a trademark of Microsoft Corporation

X Window System is a trademark of Massachusetts Institute of Technology

I-DEAS is a trademark of Structural Dynamics Research Corporation

OpenGL is a trademark of Silicon Graphics Corporation

All other brand or product names are trademarks or registered trademarks
of their respective companies or organisations.

CONTENTS

Chapter 1

Structure of the User Guide

Road Map	1-1
----------------	-----

Chapter 2

Implementation Notes

UNIX Implementation	2-1
Windows Implementation	2-9

Chapter 3

Program Philosophy

Introduction	3-1
OPERA-3d Database	3-2
Pre Processing	3-4
Post Processor	3-8

Chapter 4

Getting Started: OPERA-3d Modeller

Introduction	4-1
Building the Geometry	4-6
Creating a Background Volume	4-15
Making the Single Body Model	4-19
Building the Finite Element Mesh	4-20

Preparing the Model for Analysis	4-22
Running the Analysis	4-28
Post Processing	4-30
Displaying the Model	4-32
Checking the Solution	4-34
Field Homogeneity	4-36
Force Between Poles	4-39

Chapter 5

Geometric Modeller Features

Introduction	5-1
Picking and Hiding Entities	5-2
Display Options	5-7
Boolean Operations	5-15
Transforming Objects	5-24
Sweeping Faces	5-29
Manipulating .opc and .sat files	5-33
Mesh Control	5-34
Defining and Editing Conductors	5-40
Uses of Volume Data	5-43
Creating .op3 files	5-48
Local Coordinate Systems	5-52
Building a Library of Sub-Models	5-55
Blend and Chamfer Edges	5-57
Swift Selection of Entities	5-61

Chapter 6

Analysis Programs

Introduction	6-1
The Finite Element Method	6-2
TOSCA, Static Field Analysis	6-9
ELEKTRA, Time Varying Analysis	6-19
CARMEN, Rotating Machines Analysis	6-35
SOPRANO, High Frequency Analysis	6-49
SCALA, Space Charge Analysis	6-55

Chapter 7

Application Notes

Material Anisotropy to Represent Laminations	7-1
Combined magnetic and electric fields from TOSCA	7-3
Parameterized Models in OPERA-3d	7-6
Q and g (R/Q) factor from SOPRANO solution	7-8
External Magnetic Fields in TOSCA and SCALA	7-11
Multiply Connected Regions	7-15
Power and Energy Calculation in AC Solutions	7-26
Inductance Calculations in OPERA-3d	7-28
Use of Command Scripts to calculate Fourier Series	7-31
Complex Material Properties	7-33

Chapter 8

An Example Using the Pre Processor

Starting OPERA-3d Pre Processor	8-1
A simple model of an inductor	8-3
The Baseplane of the Model	8-4
Extending to the Third Dimension	8-15
Objects in the 3D Model	8-17
Magnetic Boundary Conditions	8-23
Displaying the 3D Model	8-26
Defining the Conductor	8-29
Examining the model with the 3d viewer	8-32
Saving TOSCA Data	8-36
Running TOSCA On-Line	8-41
Entering OPERA-3d Post Processor	8-42
Examining the Basic Solution from TOSCA	8-43
Examining the complete device from TOSCA	8-49
ELEKTRA Worked Example	8-52
Running ELEKTRA On-Line	8-58
Examining the Basic Solution from ELEKTRA	8-59
Leaving OPERA-3d Post Processor	8-66

Chapter 9

Speaker Analysis Using TOSCA

Introduction	9-1
Pre Processor	9-3
Running the TOSCA Simulation	9-39
Post Processing	9-40

Chapter 10

3D Levitation Example Using ELEKTRA

Introduction	10-1
Pre Processing, Constructing the Model	10-2
Running the ELEKTRA Simulation	10-25
Analysing the results	10-26

Chapter 11

CARMEN Example

Introduction	11-1
OPERA-3d Analysis	11-14
OPERA-3d Post Processor	11-15

Chapter 12

A Space Charge Example

Introduction	12-1
Pre Processing, Constructing the Model	12-2
Running the SCALA Simulation	12-18
Post Processing, Analyzing the Results	12-19

Chapter 13

A Cavity Example Using SOPRANO

Introduction	13-1
--------------------	------

Chapter 1

Structure of the User Guide

Road Map

The OPERA-3d User Guide is structured into the following chapters.

Implementation Notes

The OPERA-3d software can be used on both PCs and workstations, using a variety of operating systems. Each has different ways in which the software should be installed and run. This chapter outlines the differences between the systems as it applies to running the software.

Program Philosophy

An overview is given about the underlying philosophy of the software - the fact that models are created in either a geometric modeller or a pre processor including material definitions and mesh generation, and the computed results viewed and processed in the post processor.

Getting Started

A large portion of the User Guide is in this section, where a detailed description of how a model is prepared and analysed is given. New users are encouraged to spend some time going through this chapter, as it will answer many questions that can otherwise arise when using the software.

Geometric Modeller Features

The Getting Started chapter used a single worked example to show the basic operation of the OPERA-3d suite. This chapter concentrates on the use of the Modeller, with many simple worked examples illustrating the most important features of this module.

Analysis Programs

A review of the analysis programs available in the OPERA-3d suite is given in this chapter. In addition, some details are given on the finite element method and accuracy of the method, along with detailed descriptions of the techniques used in each solver.

Application Notes

This chapter contains a number of useful techniques that can be used for performing various tasks. If a question arises as to how to use the software in a particular way, this chapter should first be consulted in case an answer is presented.

Tutorials

A series of examples using the software is included. Each attempts to highlight a typical application using various analysis modules.

Chapter 2

Implementation Notes

UNIX Implementation

OpenGL Libraries

In order to run OPERA-3d pre and post processors the OpenGL run-time libraries must be present on the system. These are easily available on the standard operating system CDs or can be down-loaded from manufacturers' web sites.

The table below shows the source of the libraries, along with the version of the operating system used to compile the OPERA software.

Hardware	Operating System	OpenGL Libraries
Compaq	Tru64 Unix 5.0	Down-load from: www.service.digital.com/open3d/swps3x0.htm#KITS:True64UNIX
HP	HP-UX 11.0	Install from CD "Core OS Options: Graphics and Technical Computing Software"
IBM	AIX 4.3	OpenGL.GL32.rte is available on the AIX distribution CD
SGI	IRIX 6.5	OpenGL is part of standard IRIX installation.
SUN	Solaris 7	Install from CD "Software Supplement for the Solaris 7 Operating Environment"

Environment Variables

Setting a single Command to run OPERA

A variable **vfdir** should be set to the installation directory (for example `/u/vfopera`). OPERA-2d and OPERA-3d, including all analysis programs are then run as follows:

```
$vfdir/opera/dcl/opera.com $vfdir
```

The directory name, **\$vfdir** must be given as a parameter as shown above, so that the lower level shell files can be accessed properly. The command should be used in the definition of an alias (C-shell) or should be entered into an executable file somewhere in your search path so that a single word can be typed to start the software.

For example (C-shell):

Create an entry in *\$home/.cshrc*:

```
alias opera '$vfdir/opera/dcl/opera.com $vfdir'
```

For example (other shells):

Make sure your home directory is included in **\$path**. Create an executable file called *\$home/opera* containing the one line:

```
$vfdir/opera/dcl/opera.com $vfdir
```

Graphics Variables (Pre Processor Only)

The environment variable **VFGRAHICS** is used to set the output of the software, where the options are **SCREEN**, **FILE**, **BOTH** or **NONE**. For normal operation, this should be set to **SCREEN**. The graphics software creates an initial graphics window. There is a default size built into the programs, but this can be over-ridden by environment variables, **VFWINDOWW** and **VFWINDOWH**, which give the width and height of the window in pixels. If one of the environment variables does not exist or has a value out of range, a message giving the valid range of values is printed and the default values are used.

The graphics window is positioned at the top left of the screen. Thus the programs should be started from a terminal window at the bottom of the screen. The size and position of the graphics window can be adjusted using the window manager functions available through the mouse buttons. Subsequent pictures are scaled to fit the new window size.

***Post Processor
and Modeller***

The size and position of the program window can be adjusted using the window manager. The current settings can be saved using the menu item:

```
Windows ↓
    Save Window Settings
```

***Text and
Background***

If the environment variable **VFINV** is set to **INVERT**, the initial setting of text and background colours will be black on white instead of the default of white on black.

Starting the Software

This document assumes that the environment variables specific to OPERA have been set as explained in the previous section. In most cases it is advisable to run the programs from a suitable user directory rather than the directory containing the Vector Fields software. Similarly, it is also advisable to run the software as a “user” rather than as “system manager” since this protects against accidental overwriting of files.

Hence, as a user from a suitable local directory, OPERA may be launched by entering:–

```
opera
```

If both OPERA-2d and OPERA-3d are installed, then the system prompts for a choice and, for example, OPERA-3d could be selected:–

```
2d or 3d processing or QUIT?
3d
```

(If only one of OPERA-3d or OPERA-2d is installed, then this choice is not given).

This is followed by a list of options relating to processing environments and analysis modules available. For example, the pre processor could be selected:–

```
Option:
pre
```

If you have not set the display environment variables in your system (see earlier) then you are requested to select a method of graphics display. In this case select the screen:–

```
Graphics options: SCREen, FILE, BOTH or NONE? >scre
```

A graphics window is then opened (in addition to the text window) and control moves to the menu system. Access to the menu system is from the main menu bar at the top of the graphics window.

The command to create the terminal window can be incorporated into the commands to start the programs.

File Names and Extensions

Files created by the programs have extensions supplied by the programs. The extensions are upper-case for upper-case file names and lower-case for lower-case file names. The file name extensions are as follows. Some file extensions are common between 2d and 3d programs. For example *.bh* files.

<i>.opc</i>	Modeller data files in SAT format
<i>.opcb</i>	Binary Modeller data files including mesh
<i>.oppre</i>	pre processor data files
<i>.bh</i>	B-H data files
<i>.cond</i>	conductor data files
<i>.op3</i>	analysis database files
<i>.res</i>	log files created by analysis programs
<i>.emit</i>	emitter files used for space charge problems
<i>.comi</i>	command files
<i>.tracks</i>	particle tracking files.
<i>.table</i>	table files
<i>.ps</i>	postscript format graphics files
<i>.hgl</i>	hpgl format graphics files
<i>.pic</i>	internal picture file format

Various log files are created during the running of the pre and post processor. These are:

<i>Opera3d_Modeller_n.lp</i>	All user input and program responses to and from the modeller are recorded here
<i>Opera3d_Modeller_n.log</i>	User input to the modeller is recorded here.
<i>Opera3d_Pre_n.lp</i>	All user input and program responses to and from the pre processor are recorded here

<i>Opera3d_Pre_n.log</i>	User input to the pre processor is recorded here.
<i>Opera3d_Pre_n.backup</i>	A backup of the <i>.oppre</i> file. This is generated continuously.
<i>Opera3d_Post_n.lp</i>	All user input and program responses to and from the post processor are recorded here
<i>Opera3d_Post_n.log</i>	User input to the post processor is recorded here. These can be reused as input to the program by converting to <i>.comi</i> files

The log files are stored in a sub-directory of the project folder *opera_logs*. The file history is always retained. Each time the software is started, new *log*, *lp* etc. files are generated. Previous files are not over written. The history is retained by attaching a number, **n** to the files. The value of **n** is chosen to be the lowest available integer not already in use. This may lead to a situation where the lowest value of **n** is not necessarily the oldest file: for example if some older files were deleted, those values of **n** are re-used later.

COMI Files

The *.comi* file is a command file, that can be run using the **\$ COMINPUT** command, or using **File** → **Commands In** menu. The file is a text file, that can be created using a standard text editor, or copied from a *.log* file. The *.comi* files can also be used to automatically start up the interactive programs. Each interactive program always reads the appropriate *.comi* file on start up (although they are normally empty, the user can add commands if desired).

oppre.comi - The pre processor always reads this first.

opera.comi - The post processor always reads this first.

modeller.comi - The Modeller always reads this first.

BH Files

The directory *\$vfdir/bh* contains sample BH data files for use with OPERA-3d and OPERA-2d. The directory also includes an index *bh.index*.

Picture File Software

The directory *\$vfdir/picout* contains software for replaying picture (**.pic*) files created by the pre processor. It can be used to redisplay graphics in a picture file on the screen or to convert the file to Postscript or HPGL for printing. It is documented in the Reference Manuals under the **DEVICE** and **DUMP** commands.

Program Sizes

Extensive use is made of dynamic memory in OPERA-3d. The size (number of nodes/equations) of the analysis modules - TOSCA, SCALA, ELEKTRA, SOPRANO and CARMEN - is only limited by the available memory in the computer. This is also true for the Modeller and the postprocessor. The pre processor is the only program that has a fixed limit, which has been set sufficiently large for most user models.

A larger size of the pre processor is available in the directory `$vfdir/opera/3d/pre`. To access it, remove (or rename) the file `oppre` and rename the file `oppre_2m` to `oppre`. The larger size can create up to 2 million entities.

```
cd $vfdir/opera/3d/pre
mv oppre oppre_small
mv oppre_2m oppre
```

Swap space (Virtual Memory)

It should be noted that larger programs need larger amounts of memory. If physical memory is exceeded, swap space will be used instead. Excessive use of swap space will degrade the relative performance of the programs for large problem sizes.

Running OPERA off-line (Advanced Users)

It is sometimes beneficial to run the complete cycle of pre processing, analysis and post processing from a single command line script. It is particularly useful when large amounts of post processing are required or when a repetitive post processing task needs to be carried out. Users familiar with the Vector Fields' command language can use the off-line facility for modifying modeller or pre processor files, running an analysis, post processing and then repeating the cycle many times, all from one script.

An example of a C-shell script using the pre processor with comments is shown below:

```
#!/bin/csh
# An example of 'Off line' running OPERA-3d
# it is run by typing 'script' which should be set as executable

# When the preprocessor is run it uses commands stored in a file
# oppre.comi (and similarly the post processor uses commands
# in opera.comi)

# VFGRAPHICS is set to none, so the graphics screen is disabled.
```

```
# The commands are from the 'Command language' as defined in the
# reference manual

# START OF SCRIPT

# Set the environment variable responsible for saying what
# happens to graphics output - here we turn screen output off

setenv VFGRAPHICS none
setenv VFDIR $vfdir

# Removes old log and lp files
# files you wish to keep in this directory

rm force.op3
rm force.res
rm opera_logs/Opera3d_*. *
unalias cp

# copy the commands in prel.comi to oppre.comi, which is
# automatically run when the pre processor starts

cp prel.comi oppre.comi

# prel.comi is a VF command script which could, for example
# read in a .oppre file and generate a .op3 database

# run the pre processor using the answers to questions stored
# in 'runpre'

$vfdir/opera/dcl/opera.com $vfdir < runpre

# runpre contains for example:
# 3D
# PRE
# Q

# solve the problem (in this case the database was called
# 'force.op3')

$vfdir/opera/3d/tosca/tosca force.op3

# copy the post processing commands to the post processor
# command file which is automatically run when the post processor
# is started

cp post1.comi opera.comi

# post1.comi is a VF command script which could, for example
# read in a .op3 file, calculate some forces and dump the results
# out to another file.

# run the post processor

$vfdir/opera/3d/post/opera

# remove the opera.comi and oppre.comi files, otherwise they
# would be called when the user launched the software for
```

```
# normal operation

rm oppre.comi
rm opera.comi

# please note that when running in this fashion it is useful
# to place the command '$comi mode=cont' at the top of command

# files to stop the text window pausing when full.
```

Keyboard Entry

In the OPERA-3d Modeller and post processor, data entry through the keyboard is also available. However, input is available at any time when the Console is displayed, while the Menus also remain active. The Console window can be viewed by:

```
Windows ↓
    View Console
```

The input line of the Console is at the bottom with the (OPERA-3d > prompt), while the most recent commands are visible in the text window immediately above it. The text window can be scrolled to see commands executed earlier on in the session as well, including those entered through the Menus. Text can be *copied* and *pasted* from the text window into the Console input line or previous commands can be accessed using the arrow at the end of the line.

The Console can be hidden again by:

```
Windows ↓
    Hide Console
```

Normally, the Console is placed at the foot of the main graphics window - reducing the size of the graphical display a little. Alternatively, the Console can “float” on the screen by:

```
Windows ↓
    Undock Console
```

allowing the full graphics window to be used for display of the geometry. The Console can be returned to the bottom of the graphics window using

```
Windows ↓
    Dock Console
```

Whatever choice the user prefers (Console visible or hidden / docked or undocked) can be preserved for future Modeller sessions by:

```
Windows ↓
    Save Windows Settings
```

Windows Implementation

Licensing

All OPERA software running on a PC requires a dongle security device to be attached to one of the parallel ports on the back of the computer. Dongles are programmed at Vector Fields to allow licensed software to be run. If further licences are later purchased, Vector Fields can supply codes and instructions on how to re-programme the dongle; a new dongle is not required. Most implementations use the local (white) dongle which must be attached to the machine where the software is to be run. It is also possible to have network (red) dongles where the dongle is attached to a server PC and other PCs on this network can then use OPERA.

In order for the dongle installation to work correctly the following stages must be carried out depending on whether you have a local (white) dongle or a network (red) dongle.

Local Dongle

1. Connect the dongle to the parallel port (connect to either port if there is more than one).
2. Install the latest version of OPERA.
3. When the installation completes, the software will prompt you to install the “dongle device driver”. All users installing the software for the first time should answer “yes”. On Windows NT systems, users must have administrator privilege to perform this function.
4. Utilities are provided on the CD and the OPERA console for checking dongle status, removing the device driver and installing the device driver. The install and remove options can be run on NT only by users with Administrator privilege.

Network Dongle Server PC

1. Connect the dongle to the parallel port of the server PC (connect to either port if there is more than one).
2. Install OPERA if required to run on the server PC. (If installing on the server, answer YES to the question regarding installing the dongle device driver and go to step 4).
3. From the start-up screen of the OPERA CD, under network dongle utilities, select “Install dongle device driver”. On Windows NT systems, users must have administrator privilege to perform this function.

4. From the same screen run “Install nethasp licence manager”. This utility must be run on the server PC where the dongle is attached. This will launch the installation of a Nethasp license manager program. Select “typical installation” and the manager should be activated automatically. Comprehensive on-line help is supplied.

A range of tools is provided to help with configuring the network dongle. These can be installed from the CD (under network dongle utilities).

Network Dongle-Client PC

1. Install the latest version of OPERA if required on the client PC.
2. When the installation completes, the software will prompt you to install the “dongle device driver”. This is not necessary as the drivers reside on the server PC.
3. Start the OPERA console (see below) and select:

Options ↓

Licensing → Set Dongle Type → Network

Updating Licences

If licenses need to be updated (extra software is purchased for example), this can be carried out by the user, and a new dongle is not required.

1. Start OPERA
2. From the main OPERA Console (see below) select:

Options ↓

Licensing → Update Dongle Licensing...

3. Select **Read Codes From File** and select the file containing the new codes. The file should be in plain text format.

The OPERA Console

The OPERA Console is started from the menu bar as follows:

Start → Programs → Vector Fields OPERA → OPERA 8.5

Alternatively the console can be started from the system icon tray.

Click on the blue and white VF icon as shown here: .

The console is the ‘navigation centre’ for the complete suite of OPERA software. It allows the following operations:

- Setting of 2d and 3d project directories
- Launching the 2d and 3d pre and post processors
- Launching analyses and organising batch runs
- Viewing analysis result log (*.res*) and emitter (*.emit*) files
- Changing the CPU priority for an analysis
- Listing and updating licensing
- Adjusting some windows parameters, including reversing the foreground and background colours, adjusting the Graphics Window size and the Text IO history buffer.

All options will be shown in the console menus, but only licensed modules will be available. Contact Vector Fields for licensing information. Using the mouse right button on the system tray VF icon will bring up a similar range of options to those available on the menu bar.

When starting the 2d or 3d pre/post processors for the first time, a project folder will be requested. This can be changed at any time, but you will not be prompted when you subsequently start the software. This is the default folder in which OPERA will be working.

Running OPERA Pre and Post Processing

OPERA-3d Modeller

The OPERA-3d modeller is started from the console menu bar:

OPERA-3d ↓
Modeller

The project or working folder can be changed from the default:

OPERA-3d ↓
Change Project Folder → OPERA-3d ...

The dialogue box will then allow you to browse your computer’s folder structure or enter a path name directly. A new folder name can be added if required.

OPERA-3d Pre Processor

The OPERA-3d pre processor is started from the console menu bar:

```
OPERA-3d ↓
  Pre-Processor
```

The project or working folder can be changed from the default:

```
OPERA-3d ↓
  Change Project Folder → OPERA-3d ...
```

The dialogue box will then allow you to browse your computer's folder structure or enter a path name directly. A new folder name can be added if required.

OPERA-3d Post Processor

The OPERA-3d post processor is started from the console menu bar:

```
OPERA-3d ↓
  Post-Processor
```

The project or working folder can be changed from the default:

```
OPERA-3d ↓
  Change Project Folder → OPERA-3d ...
```

The dialogue box will then allow you to browse your computer's folder structure or enter a path name directly. A new folder name can be added if required.

Running OPERA Analysis Modules

Interactive Solutions

All the analysis options are now set in the pre processing. It is possible to run interactive analyses directly from the modeller or pre processor. To interactively run from the console menu bar, the **OPERA-2D** or **OPERA-3D** analysis modules are accessed as follows:

```
OPERA-2d/3d ↓
  Interactive Solution ...
```

The appropriate analysis module and *.op2* or *.op3* file are then selected

Once an analysis is started, a window will appear indicating the various stages of the solution process. When complete a Windows message will appear asking if you wish to close the solver window. This gives you the opportunity to scroll through the solution steps if you wish. Alternatively you can view the saved *.res* file for similar information. It is usually a good idea to view the *.res* file as any

run-time problems will have been listed here. The file is viewed with the default editor (notepad) as follows.

```
File ↓  
    Display Res File ...
```

The default editor can be changed from the console menu:

```
Options ↓  
    Change Editor ...
```

Batch Solutions

Data can be placed in batch files for later analysis. Many separate analyses can be added to a batch file for overnight or weekend runs if required. To select a file for running in batch:

```
OPERA-2d/3d ↓  
    Add to batch queue ...
```

Once all the required files have been added, the batch file can be started:

```
Batch Process ↓  
    Start Batch Analysis ...
```

Options to clear or list the batch queue are also available under this menu item.

Running OPERA-3d off-line

A utility program called *winbat.exe* has been included with OPERA to allow the automatic execution of analyses from simple command scripts. This removes the need to interact with the OPERA console if a large number of analyses with post processing need to be carried out. The program *winbat.exe* is run from a command/DOS prompt and requires the name of the file containing the command script as a parameter. For example:

```
winbat.exe script.batch
```

The command script contains a series of commands, that use an identical syntax to normal DOS batch commands. To see the syntax type **help command** in a DOS window. The commands available are listed below:

```
copy, del, move, rem, dir, for, mkdir, rmdir, echo, cd,  
set.
```

The **cd** and **set** commands are simple implementations only, without the full features available.

All standard executable files ending with the extension *.exe*, *.bat* and *.com* can be executed from the script. Specifying the full path to the executable is advisable. Paths or filenames that include spaces must be enclosed in quotes “ ”.

The program *winbat.exe* can be executed with a *-n* switch. This will prevent the completion message being displayed, and is useful if the winbat program is called from within a script.

Example

An example script to run the pre processor followed by TOSCA analysis module and finally the post processor is shown below:

```
REM Set some variables
REM Set installation folder
set VFBATCH=C:\program files\Vector Fields\opera 8.5
REM Set local folder
set LOCALDIR=C:\opera\work1
REM Make sure we are in the correct working folder
cd %LOCALDIR%
REM set the comi file for running the pre-processor
REM This file contains the commands necessary to
REM build the model
REM and create the new database file: mydata.op3
copy mypre3d.comi oppre.comi
REM launch the pre-processor
REM /local runs in current folder
REM /min runs iconised
"%VFBATCH%\pre\pre.exe" /local /min
REM Run the analysis on the OP3 file created by the
REM Pre-Processor command script
"%VFBATCH%\solvers\3d\tosca.exe" "%LOCALDIR%\mydata3d.op3"
REM prepare the post-processing command file
copy mypost3d.comi opera.comi
REM launch the Post-processor
"%VFBATCH%\post\post.exe" /local /min
REM clear out the command input files and the op3 database.
del oppre.comi
del opera.comi
del mydata3d.op3
```

Please note that for users running Windows NT, 2000 and XP, it is not necessary to use *winbat.exe*. Batch scripts as above can be run directly. If running directly under NT, 2000 or XP the **cd** command should be replaced with **cd /D**

Text IO Window

The OPERA pre processor allows for data entry via the keyboard. This is achieved by selecting **MENU OFF** from the top menu bar. A text input window prompts the user to enter the appropriate commands. This window may be moved or resized just like any other window. Its contents may be scrolled, either by using

the scroll bar, or with the <up-arrow>, <down-arrow>, <page-up> and <page-down> keys. If the prompt is out of sight, pressing the <return> or any other character will scroll the window back to the prompt.

The line currently being entered may be edited; the <Insert> key toggles between overstrike and insert mode, with the cursor prompt changing to indicate the mode. The <left-arrow>, <right-arrow>, <home> and <end> keys may be used to move the cursor prompt. Deletion can be done with the <backspace> key.

Previous commands that have been executed may be re-selected by pressing <shift up-arrow> or <shift down-arrow>. These commands can be edited as described above and then re-issued with the <return> key.

The number of lines in the Text IO history buffer can be controlled using:

```
Options ↓
    Window Preferences
```

and set the value of **Message Screen History**.

The Text IO window can be hidden or 'always on top' using the top (grey) menu bar command:

```
View ↓
    Console
```

Typing ^ (normally SHIFT+6) at the cursor prompt will restore control to the menus.

Keyboard Entry

In the OPERA-3d Modeller and post processor, data entry through the keyboard is also available. However, input is available at any time when the Console is displayed, while the Menus also remain active. The Console window can be viewed by:

```
Windows ↓
    View Console
```

The input line of the Console is at the bottom with the (OPERA-3d > prompt), while the most recent commands are visible in the text window immediately above it. The text window can be scrolled to see commands executed earlier on in the session as well, including those entered through the Menus. Text can be *copied* and *pasted* from the text window into the Console input line or previous commands can be accessed using the arrow at the end of the line.

The Console can be hidden again by:

Windows ↓
Hide Console

Normally, the Console is placed at the foot of the main graphics window - reducing the size of the graphical display a little. Alternatively, the Console can “float” on the screen by:

Windows ↓
Undock Console

allowing the full graphics window to be used for display of the geometry. The Console can be returned to the bottom of the graphics window using

Windows ↓
Dock Console

Whatever choice the user prefers (Console visible or hidden / docked or undocked) can be preserved for future Modeller sessions by:

Windows ↓
Save Windows Settings

File Names and Extensions

Files created by the programs have extensions supplied by the programs. The extensions are upper-case for upper-case file names and lower-case for lower-case filenames. The file name extensions are as follows. Some file extensions are common between 2d and 3d programs. For example *.bh* files.

3D Files

<i>.opc</i>	Modeller data files in SAT format
<i>.opcb</i>	Binary Modeller data files including mesh
<i>.oppre</i>	pre processor data files
<i>.bh</i>	B-H data files
<i>.cond</i>	conductor data files
<i>.op3</i>	analysis database files
<i>.res</i>	log files created by analysis programs
<i>.emit</i>	emitter files used for space charge problems
<i>.comi</i>	command files
<i>.tracks</i>	particle tracking files

<i>.table</i>	table files
<i>.ps</i>	postscript format graphics files
<i>.hgl</i>	hpgl format graphics files
<i>.pic</i>	internal picture format graphics files

Various log files are created during the running of the pre and post processor. These are:

<i>Opera3d_Modeller_n.lp</i>	All user input and program responses to and from the modeller are recorded here
<i>Opera3d_Modeller_n.log</i>	User input to the modeller is recorded here.
<i>Opera3d_Pre_n.lp</i>	All user input and program responses to and from the pre processor are recorded here.
<i>Opera3d_Pre_n.log</i>	User input to the pre processor is recorded here. These can be reused as input to the program by converting to <i>.comi</i> files.
<i>Opera3d_Pre_n.backup</i>	A backup of the <i>.oppre</i> file. This is generated continuously.
<i>Opera3d_Post_n.lp</i>	All user input and program responses to and from the post processor are recorded here.
<i>Opera3d_Post_n.log</i>	User input to the post processor is recorded here. These can be reused as input to the program by converting to <i>.comi</i> files.

The log files are stored in a sub-directory of the project folder *opera_logs*. The file history is always retained. Each time the software is started, new *log*, *lp* etc. files are generated. Previous files are not overwritten. The history is retained by attaching a number, **n** to the files. The value of **n** is chosen to be the lowest available integer not already in use. This may lead to a situation where the lowest value of **n** is not necessarily the oldest file: for example if some older files were deleted, those values of **n** are re-used later.

COMI Files

The *.comi* file is a command file, that can be run using the **\$ COMINPUT** command, or using **File** → **Commands In** menu. The file is a text file, that can be created using a standard text editor. The *.comi* files can also be used to automatically start up the interactive programs. Each interactive program always reads the appropriate *.comi* file on start up (although they are normally empty, the user can add commands if desired).

oppre.comi - The pre processor always reads this first.

opera.comi - The post processor always reads this first.

modeller.comi - The Modeller always reads this first.

BH Files

The *bh* folder in the installation folder contains sample BH data files for use with OPERA-2d and OPERA-3d. The folder also includes a *bh.index* file.

Windows Menus

In addition to the standard GUI menus in the pre processor, there is a set of additional menus specifically for the Windows implementation. Their function is to control the behaviour of the OPERA pre processor within the Windows environment.

FILE Menu

The **FILE** menu allows printing of the windows, and displaying some related files.

Selecting **FILE** → **Print** allows printing of all the windows, or just the graphics window. If **All Windows** is selected, then the surrounding border and the OPERA GUI menus are included in the print. If **Graphics Window** is selected, the top level menus are omitted (but any pull-down menus overlapping the graphics window are included in the print). The Windows print manager is used to perform the printing in the usual way. If the 3d-Viewer window has been activated, then a third option **FILE** → **Print** → **3d-Viewer** is available, which just prints the contents of the 3d-Viewer window. The Windows print manager is used to perform the printing in the usual way.

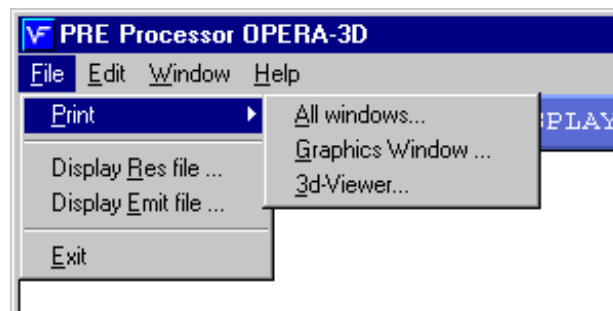


Figure 2.1 File menu including Print sub-menu

The option **FILE** → **Display Res File** launches Windows Notepad and prompts for the file to be loaded. All files with the *.res* extension are displayed for selection.

The option **FILE** → **Display Emit File** is similar to above, with files having extension *.emit* displayed for selection.

EDIT Menu

The **EDIT** menu copies the contents of the windows to the clipboard. As above, the options are for **All Windows** or just the **Graphics Window** to be copied. The **3d-Viewer** options copies the 3d-Viewer window only to the clipboard.

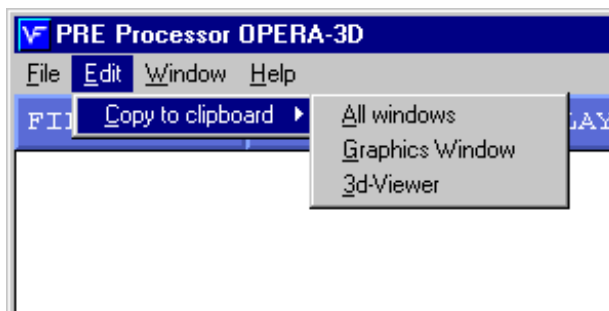


Figure 2.2 Edit menu allowing copying to clipboard

WINDOW Menu

The **WINDOW** menu controls the positioning of the Console and the Graphics windows on the screen.

The options for the Console window are:

Normal: The Console window (showing the equivalent keyboard commands on menu selection, or allowing keyboard input of the appropriate commands in place of the GUI) is on top when interacting with it, but is positioned behind the graphics window when the GUI is active.

Always on Top: The Console window is forced to always be positioned on top of all other windows.

Hidden: The Console window is deleted completely, but can be displayed again using one of the above options.

The options for the Graphics window are:

Restore: The Graphics window is reduced in size, and can then be resized as required in the normal manner.

Maximised: This sets the Graphics window to the default size, where it completely fills the available space within the OPERA window.

In addition, the option **WINDOW** → **3d-Viewer** allows the 3d-Viewer window to be restored if it is not visible (provided it has been activated).

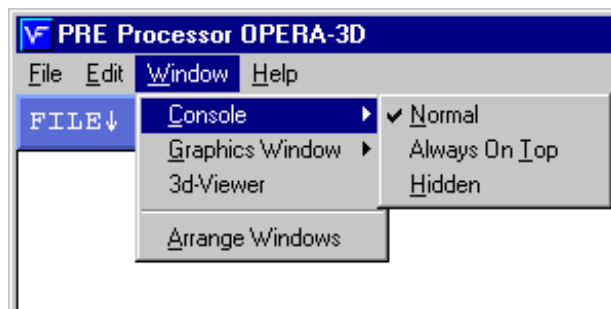


Figure 2.3 Window menu controlling positioning of Console, Graphics and 3d-Viewer windows. The Console window options are shown.

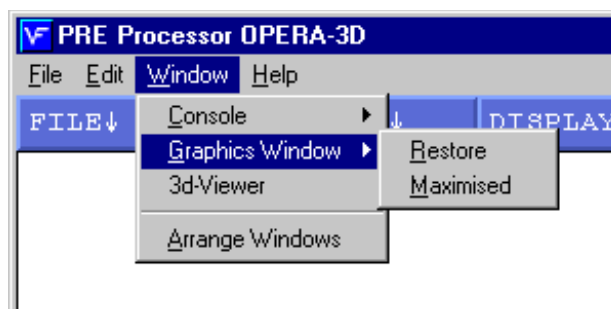


Figure 2.4 Window menu controlling positioning of Console, Graphics and 3d-Viewer windows. The Graphics window options are shown.

A further option for displaying the Graphics and 3d-Viewer windows is **WINDOW → Arrange Windows**, which has the effect of equally partitioning the display area for the two windows.

HELP Menu

The **HELP** window allows access to the online help. The relevant Reference Manual is present in WinHelp format. In addition, the **ABOUT** option displays the current software version number, and the contact details for Vector Fields.

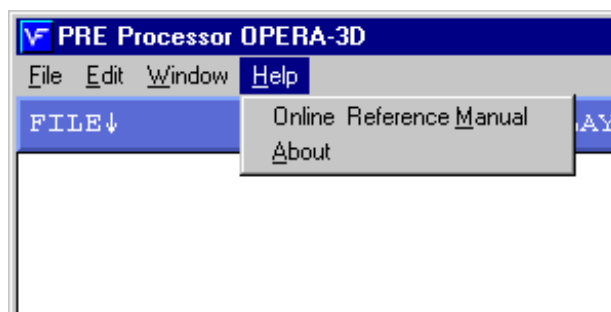


Figure 2.5 Help menu allowing online help to be displayed (WinHelp)

Chapter 3

Program Philosophy

Introduction

The aim of this user guide is to provide simple examples to introduce some of the features of OPERA-3d.

OPERA-3d is the pre and post processor for the well known electromagnetic analysis programs including TOSCA, ELEKTRA, CARMEN, SOPRANO and SCALA.

OPERA-3d contains two modules to generate data files and models for the electromagnetic analysis programs. The Modeller uses geometric primitive volumes and Boolean operations to construct the model while the pre processor uses extrusion from a 2D cross-section. Both modules allow the user to create the finite element mesh, specify conductor geometry, define material characteristics including, for example, non-linear and anisotropic descriptions, and graphically examine and display the data.

Similarly, the OPERA-3d post processor provides facilities necessary for displaying the electromagnetic fields. As well as displaying field quantities as graphs and contour maps, the OPERA-3d post processor can calculate and display many derived quantities and can plot particle trajectories through the calculated fields.

The Reference Manual gives detailed information on the commands used in OPERA-3d.

OPERA-3d Database

OPERA-3d uses a binary database, much of which is transparent to a user of the software. Those aspects that are relevant to the use of the software will be outlined in the following sections. In particular, either the Modeller or the pre processor can be used to create a binary database (*.op3*) directly, which is accessed by all the analysis modules and the post processor.

Simulations

The structure of the database is based on the principle of storing the geometric data for a model, and then creating a number of “simulations”. The database created is entirely self contained, and does not rely on additional data files (BH tables etc. are all incorporated into the database). Also the database is not tied to a single analysis module. (Note that SCALA is slightly different as it relies on an external emitter file).

Running a simulation consists of using the geometry stored in the database, and running one of the analysis modules. Sufficient data to direct the solution (including which analysis module should be used) is stored in each simulation section. The results are then stored in the database for that simulation, for future post processing.

Each simulation can utilise a different analysis module, so that one database for example could contain the geometry, followed by a TOSCA simulation, an ELEKTRA-SS simulation, and then a series of ELEKTRA-TR simulations.

The database has a single set of units applied to it, and this is used for each simulation. It is not possible to have different units for each simulation in the same database.

Multiple Drives

All coils and boundary conditions are labelled. Each coil must be given one of these labels, although a number of coils can be given the same drive label. Each set of coils is then assigned a drive type (in ELEKTRA-TR and CARMEN), and phase (in ELEKTRA-SS and SOPRANO-SS).

Boundary conditions may also have a drive label (although this is not compulsory), allowing different drive functions and phases to be allocated to groups of boundary facets.

If the same drive label is used for both boundary conditions and coil definitions, then the same drive type will be assigned to both.

Material Properties

In all analysis modules (except SOPRANO and SCALA), material non-linearity and anisotropy is available. The BH curves are stored in the database itself. This implies that once the database is created, any changes to the BH data file will have no impact on the simulation, as it is the BH curve stored in the binary database that is actually used by the simulation (see comments above about database being self contained).

When defining material properties in the pre processor, three vector quantities may be defined: source current density (in ELEKTRA and SOPRANO), velocity (in ELEKTRA-VL), and material orientation (to define anisotropy). Each of these can also be functional (i.e. a function of position).

NB: *Since source current density and velocity vectors were introduced as extra items in Version 7, it follows that pre processor files prior to Version 7 are no longer compatible. It will be necessary to define source current density and velocity again for these solution types.*

Restarts

It is possible to restart an analysis, starting from any simulation held in the database, not necessarily the last one (although a restart must be from a simulation of the same type as the original).

For example, the ELEKTRA-TR module can be restarted from any previous solution currently stored, and need not be the last solution present. It is possible then to run a simulation in ELEKTRA-TR until a certain time. A restart can then be carried out starting from an earlier time, and using a smaller time step. This could be used for example to check the accuracy of the solution, by comparing the results at a particular time arising from using two different time steps over part of the analysis.

Since coil fields are grouped (see section “Multiple Drives” on page 3-2), a restart will use the coil fields already created from an earlier simulation – they are not re-computed. This will greatly improve the efficiency when complex conductor sets are being used.

Pre Processing

The software package OPERA-3d includes the facility for automatic tetrahedral mesh generation. This feature proves extremely convenient for many problems, by greatly improving the ease with which models can be built. This is the only type of meshing available from the Modeller.

In the pre processor, it is also possible to build models in terms of brick (hexahedral) elements, and indeed previously defined models can continue to be processed in the OPERA-3d suite through this route.

Some guidelines are included below to aid in the decision as to which form of element to use – whether to use hexahedral meshes as before, or whether to make use of the automatically generated meshes using tetrahedral elements.

Accuracy

One important aspect in making this decision is the accuracy of results that are to be obtained. It is a feature of the two types of finite element (irrespective of how they are created in the first place) that, for the same number of nodes, hexahedra can give greater accuracy than tetrahedra – hence the reason for maintaining the ability to produce hexahedral meshes within OPERA-3d.

In general for highest accuracy, it is necessary to align the element edges with the equi-potential lines. This can be achieved with hexahedra, but automatically generated tetrahedra edges are unlikely to ever align correctly, so errors are introduced.

For problems requiring high accuracy, hexahedral elements must be used therefore. An example of such an application would be MRI magnets, where exceptionally high accuracy in the field solution is required. Many more tetrahedral elements would probably be needed to achieve the same accuracy. Of course, the increased solution time associated with the larger problem must be weighed against the time to create the model using only hexahedral blocks.

Another instance where hexahedra should be used is when large, nearly balanced forces, are to be computed. It is then necessary to have elements having a very regular shape and distribution, to avoid as much of the “discretisation error” as possible. This can only be achieved by having strong control over the shape and position of the elements, as is the case with the hexahedral mesh generation strategy within OPERA-3d.

Ease of Modelling

Where accuracy is less of an issue, but ease of meshing is paramount, it is recommended that the automatic mesh generator is used. For example in motors and generators, which have very complex geometries, it is not necessary to have the high levels of accuracy as in the previous examples. On the other hand, the complex geometric shapes make modelling in terms of hexahedra very time consuming. This is an ideal situation for using tetrahedral elements, created automatically by the Modeller or pre processor.

Where improved accuracy is required, for example in the air gap where the torque calculations are to be carried out, it is suggested that quadratic elements are used locally. All the analysis solvers (apart from SOPRANO) allow mixed linear and quadratic elements in same problem, and are recommended for locally improving the accuracy within a model.

Qualifications on Use of Automatic Meshing

Pre Processor

In the pre processor, there is the concept of a ‘base plane’, which can be created using general polygons. The base plane is then extruded to form volumes. It is necessary to create the mesh within the pre processor. This is carried out in two stages – first a surface mesh is created, followed by a volume mesh.

If only 3 or 4-sided facets are used on the base plane, then it is possible to direct the surface mesh to be quadrilateral (and hence generate a hexahedral volume mesh). If a polygon facet is used anywhere on the base plane, then it will only be possible to create a triangular surface mesh, and consequently the volume mesh will be based on tetrahedra.

Having made this general statement, some qualifications should also be made. If 3 or 4-sided facets are created (not polygons), it is possible to modify the sub-divisions so that they are irregular. It will then NOT be possible to create a quadrilateral surface mesh.

Conversely, if 4-sided polygons are defined, with a regular sub-division, it may still be possible to create a quadrilateral surface mesh. Using the **CHECK** command confirms if the subdivision is regular, and hence whether a quadrilateral surface mesh can be generated. If this is the case, then subsequent modifications to the subdivision will retain its regularity (unless an irregular subdivision is specifically assigned).

An important restriction to note at this point is that if a model constructed with the pre processor uses periodicity in TOSCA or SCALA, then it is necessary that

either hexahedral elements are created, or a regular tetrahedral subdivision is used on the periodic faces.

Another restriction is that if a polygon contains more than 4 vertices, all points must be coplanar. The software does not allow for other types of surface, and will assume the points are in a plane.

Modeller

In the Modeller, the user is able to construct the model using primitive volumes, swept surfaces and Boolean operations. Unlike the pre processor, the resulting volumes are not generally shapes that could be created through an extrusion process. Consequently, regular meshing with hexahedra is not available.

Like the pre processor, mesh generation in the Modeller is a two stage process. Surfaces of volumes (cells) are initially discretised into triangles. Controls are available to define the exactness of the representation of curved surfaces. Then, using the surface mesh, each cell is meshed automatically into tetrahedra.

Element size can be controlled by defining a maximum element size on vertices, edges, faces or cells within a model. This allows the mesh to be concentrated in areas of interest, where high accuracy is required, or where the field is changing rapidly.

Note that for TOSCA and SCALA models in which periodicity boundary conditions are required, the Modeller can automatically ensure that the mesh on the boundaries is identical, unlike the pre processor (see [“Pre Processor” on page 3-5](#)).

Functional Boundary Conditions

A useful feature is the ability to see the functional variation of boundary conditions in the Modeller and pre processor. A contour map can be displayed over the boundary, displaying the variation of boundary values.

Mesh Continuity

It must be stressed that inside a model the mesh has to be continuous. No holes or gaps in the mesh are allowed as they have the effect of an external boundary inside the mesh. In the Modeller, this is mostly avoided automatically as the internal description of the geometry usually ensures that neighbouring cells are using the same surface definition. Of course, if the user actually defines two volumes with slightly different coordinates then a gap or overlap will occur. There are tools

within the Modeller to assist the user in “healing” such overlaps to remove very small volumes and surfaces.

In the pre processor, the user should ensure that each edge on the base plane is used by two polygons, unless it is on a physical boundary (possibly internal) of the model. Furthermore, the regions of the base plane must not overlap. There is a **CHECK** command which performs a check of the mesh, counts the nodes and labels all external facets with the label **EXTERNAL**. After having run the **CHECK** command you can select all external facets and display them. If there are external facets inside the model, it is an indication of a modelling error.

To model a complicated geometry with the pre processor, a mesh can be built with several different meshes. However, the user is responsible for the mesh matching at the interface between two meshes. As previously mentioned, the resulting mesh has to be continuous.

Post Processor

System Variables

System variable names are chosen to help identify the nature of the variable. This reflects the fact that real and imaginary values are available simultaneously in the post processor for ELEKTRA-SS and SOPRANO-SS results. The magnetic field values are available as **BX**, **BY** and **BZ** (instantaneous values at the selected time), along with **RBX**, **RBX**, **RBZ**, **IBX**, **IBY**, **IBZ** (which are the real and imaginary parts of the field components).

NB: *Prior to v7.5 of the post processor, the imaginary part of a component was defined as its value at time $\omega t = 90^\circ$. From v7.5, this has been modified to be the value at time $\omega t = -90^\circ$. This brings the post processor into line with common engineering convention for leading and lagging phase angles expressed as complex numbers.*

For each simulation, a subset of system variables are loaded, depending on the simulation type. The system variables loaded for an electrostatic TOSCA simulation will be different to those loaded for a magnetostatic TOSCA simulation (i.e. electric field rather than magnetic field will be loaded, and the potential will be named *V* in the electrostatic simulation).

Other system variables will be available *on request*. For example, the error values from a TOSCA magnetostatic analysis **ERRB** (see [page 3-9](#)) is not automatically available when the database is activated and loaded. The user must add it to the list of available system variables, selecting appropriate units.

This feature is particularly useful if the user wishes to make new system variables available that do not form part of the existing analysis results. For example, if an electrostatic TOSCA solution is loaded into the post processor, a magnetic field from another simulation can be added to the database using the **TABLE** command. The electrostatic variables are the default set, but the magnetic system variables are also available to view the magnetic fields. These must be loaded manually by the user, if for example **BX** is to be displayed or used for tracking particles in combined magnetic and electric fields.

For coil only problems, the default variable set of **HX**, **HY**, **HZ**, **BX**, **BY** and **BZ** are available. For current flow problems, the variables **HX**, **HY** and **HZ** are available following a request for **INTEGRAL FIELD** computations. The variables **HX**, **HY** and **HZ** are disabled when selecting **NODAL FIELD** computations again.

Nodally Averaged Fields

All the solutions available in the database are ‘nodally averaged’. A smoothing process is used to ensure continuous fields within each different material volume. Between different materials, those components that should be discontinuous are allowed to be so.

The options for displaying field quantities are **NODAL** and **INTEGRAL** (the **SHAPE** option prior to v7 is not available).

Labels from the Pre Processor

To help select surfaces and volumes in the post processor for displaying/computing quantities, it is possible to make use of additional labels assigned in the Modeller or pre processor. These labels are stored in the database, and can be used later in the post processor. For example, different parts of the boundary can be assigned different labels, and hence selected independently in the post processor.

Solution Accuracy

The solutions obtained using finite element methods (as used in OPERA-3d) are numerical approximations to the real solution to the continuum physics equations. The accuracy of the approximation will be dependent on the ability of the finite element basis function (See “[The Finite Element Method](#)” on page 6-2.) to represent the local spatial variation of the vector or scalar potential. Consequently, there will be a local error associated with each finite element of a mesh.

The OPERA-3d analysis programs compute the upper bound of the local error in the derived fields - magnetic flux density, electric flux density and current density - as follows:

Table 1: Computed error in derived fields

Derived field error	Analysis program
Magnetic flux density	TOSCA magnetostatics ELEKTRA CARMEN
Electric flux density	TOSCA electrostatics, SCALA
Current density	TOSCA current flow

using an *a posteriori* method based on comparison of field values obtained from differentiation of the element basis functions and the nodally averaged fields computed in the analysis.

The local error can be displayed in the post processor using vector system variables **ERRB**, **ERRD** or **ERRJ**, dependent on the analysis type. These system var-

ables are not loaded automatically, but must be loaded by the user specifically. The x , y and z components of each error can be accessed e.g. **ERRBX** for the x component of the local error in the magnetic flux density and the values are given in the same units as the derived field on which the error has been computed.

SOPRANO uses edge elements to solve for the electric field directly. Consequently, the error computation algorithm cannot be applied to SOPRANO solutions and error vectors are not yet available in the post processor. Likewise, they are not yet available in SCALA.

Chapter 4

Getting Started: OPERA-3d Modeller

Introduction

In this chapter, the most important concepts for pre processing, analysing and post processing of OPERA-3d models are introduced. Building a geometry using the Modeller, exploiting the symmetry to reduce the size of problem solved and generating the finite element mesh are covered. Setting up the analysis by choosing appropriate material characteristics, solution type and solver module is also discussed. Finally, the post processor is used to obtain results from the solved database.

The Model

The model to be created in this introduction is shown in Figure 4.1. It represents a simplified permanent magnet MRI (Magnetic Resonance Imaging) system. The *blue* material is a permanent magnet, magnetised to produce a uniform field in the region between the *purple* magnet poles. The poles are made from good quality steel and include shims of the same steel (annular rings added to the pole face) to improve the field quality. The frame (in *green*), constructed from a lower quality steel, acts as the return path for the flux.

Because of the symmetry of the model, it is sufficient to solve the magnetic fields using only one eighth of the geometry of the MRI system, with a surrounding free space region. However, a half model of the geometry will be constructed initially as this is easier. Figure 4.2 shows the half model, which represents the upper pole and magnet and the top half of the frame. The symmetry of the one eighth model will be implied by appropriate boundary conditions. The model is constructed in CGS units.

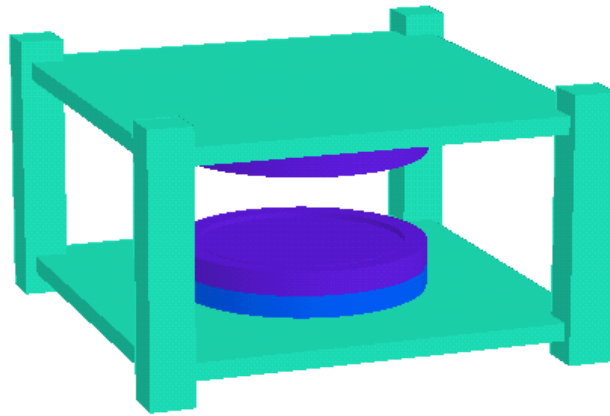


Figure 4.1 The MRI magnet geometry

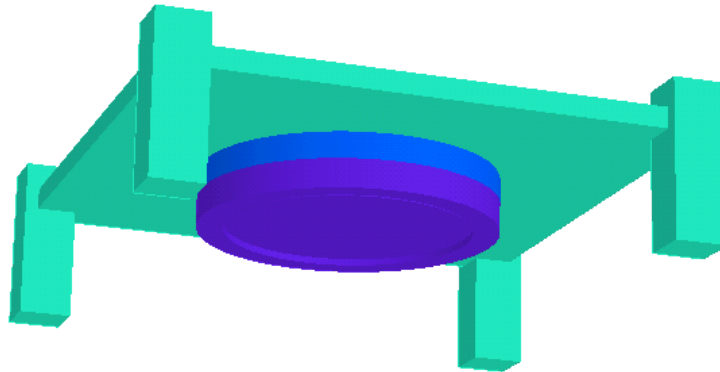



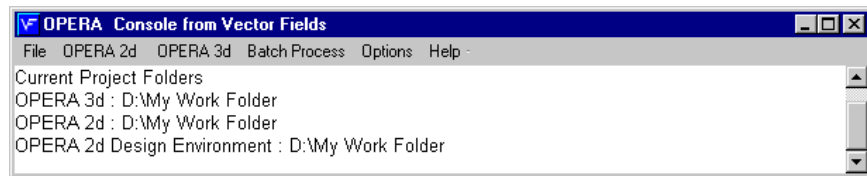
Figure 4.2 The half geometric model to be constructed

Mesh quality is important in MRI applications and several controls will be used to improve the mesh. The analysis will be non-linear requiring the user to specify magnetic characteristics (\mathbf{B} v \mathbf{H} curves) for the magnet and the steel. In the post processing, the magnetic flux density in the geometry will be examined, the homogeneity of the dipole field determined using Legendre Polynomial fitting and the force between the poles calculated.

Starting the Modeller

Microsoft Windows Platforms

The Modeller is started from the OPERA Console window. To start the Console window, click on the VF Icon in the system tray, , or use the Start menu to locate Vector Fields OPERA under Programs. The Console window presents a menubar.



Click on **OPERA-3d** and select **Modeller** from the menu.

Unix Platforms

Enter

```
$vfdir/opera/dcl/opera.com $vfdir
```

where **\$vfdir** is the directory in which the OPERA software has been installed.

If both OPERA-2d and OPERA-3d have been installed, the user will be further prompted

```
2d or 3d processing or QUIT?
```

Reply

```
3d
```

The program will then offer the user a selection of modules. Enter

```
modeller
```

Using the Modeller

When the Modeller starts, the user is presented with a display showing the 3D axes (Figure 4.3). Note that the display will look slightly different on a Unix platform, but the functionality will be the same.

A selection of toolbars at the top of the screen is available to control the Modeller. The toolbars contain icons, which perform frequently used operations. The top menubar activates pop-up menus, which allow all operations to be accessed. The Modeller can also be controlled using a keyboard entry. The user is free to switch between toolbar/menubar and keyboard modes at any time. The console for keyboard entry is activated by clicking with the left mouse button on the **Windows** menu on the top menubar, and selecting **View console**. Figure 4.4 shows the Modeller screen when the keyboard entry console window has been activated. The same menu can be used to remove the console window by selecting **Hide console**. To enter keyboard commands at the console window, move the mouse into the command entry line at the bottom of the screen (prefixed by the **OPERA-3d >** prompt) and click the left mouse button. A flashing cursor appears and the user may type commands.

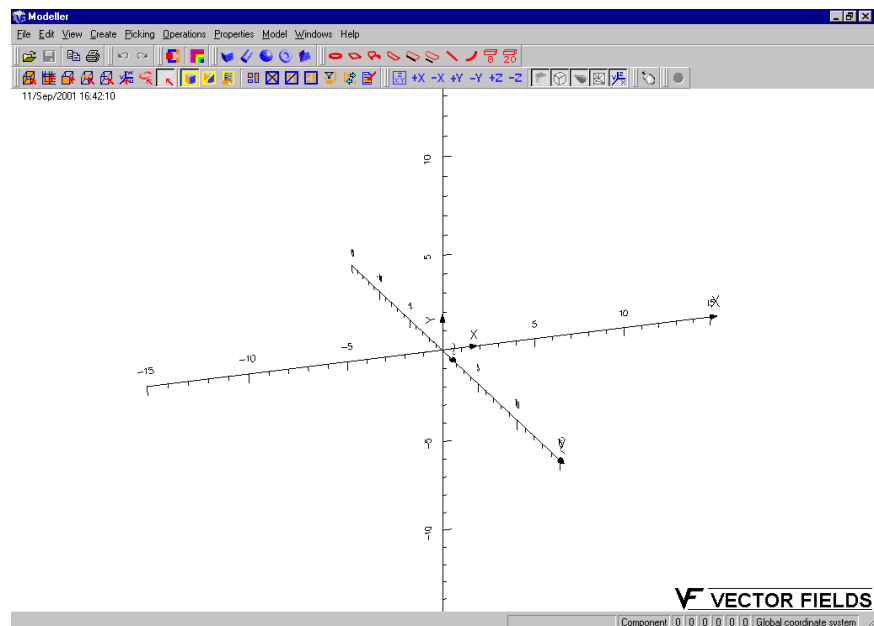




Figure 4.3 The initial display using Microsoft Windows

Other important features of the user interface are **Undo**  and **Redo** . These allow the user to return the Modeller session to any place in its history, for example to correct an incorrectly given instruction or to create several variations from a base model. The features may also be accessed from the **Edit** menu or via the keyboard.

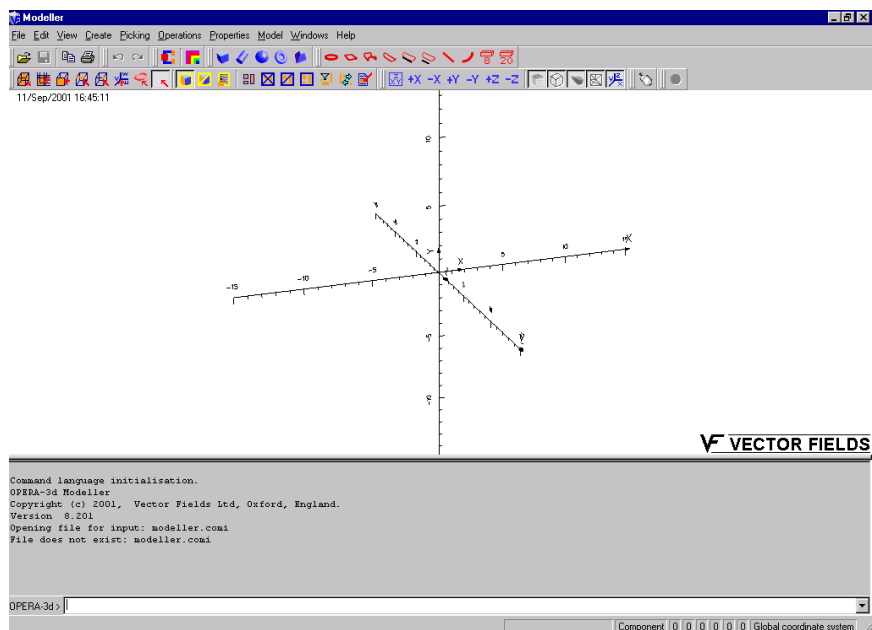


Figure 4.4 Console window activated

Building the Geometry

The Permanent Magnet

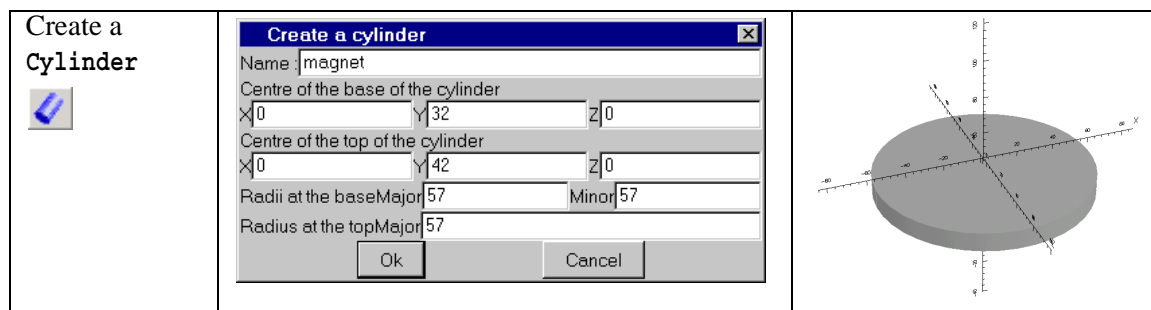
The first part of the geometry entered is the magnet. This is constructed from a single cylinder, has a radius of 57 cm and a thickness of 10 cm. The circular face of the magnet touching the pole is 32 cm from the symmetry plane, which will be assumed to be at $Y = 0$ cm. Geometric entities can be created in the Modeller by either method - picking the appropriate icon from the toolbar or entering a keyboard command.

Using an Icon

To create the cylinder from an icon, click on the blue cylinder icon on the toolbar using the left mouse button. This brings up a dialog box to define the cylinder (note that cylinder is a generic name - the options allow construction of cones and elliptic cylinders also).


To enter data in a dialog box, either select the field using the mouse, or move to the field using the tab button on the keyboard. Pressing the **Enter** key is equivalent to selecting **OK** on the dialog box. The user may escape from any dialog box without making any changes to the model by using the **Esc** key (equivalent to clicking on **Cancel**).


The cylinder is defined by completing the dialog box as shown and clicking on **Ok**. This produces the disk shown in the attached thumbnail figure.








Controlling the View

The cylinder can be viewed from many positions. Move the mouse into the graphics window, hold down the left mouse button and move the mouse. This allows the view of the cylinder to be rotated. Now hold the right button of the mouse

down instead and move the mouse. This gives translation allowing the object of interest to be moved to the centre of the screen. For users with a three-button mouse, holding the centre button allows the view to be zoomed in and out. Users with a two button mouse can toggle the function of the right button between translate and zoom by clicking on the mouse icon .

If the user modifies the view in such a way that it is not easy to return to an overall view of the geometry, the view can be re-initialised by selecting the **Initial View** icon from the toolbar .

This will adjust the view so that all the visible geometry can be seen. The other icons on the toolbar also control the view. The geometry can be viewed along any of the major axis directions (, , etc.). Two icons are available to toggle between solid view  on and off and wire-frame view  on and off.

The axes can also be toggled on or off .

These functions can also be accessed through the **View** menu on the top menubar.

Properties of the Magnet


The default properties of the cylinder that has been called magnet are:

Material **AIR** ($\mu = \mu_0$, $\varepsilon = \varepsilon_0$, $\sigma = 0$)



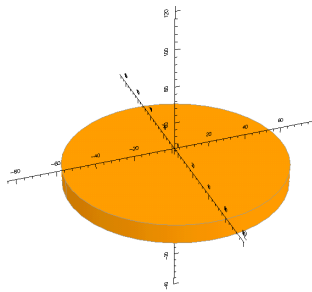
Potential type **REDUCED**

Element type **LINEAR**

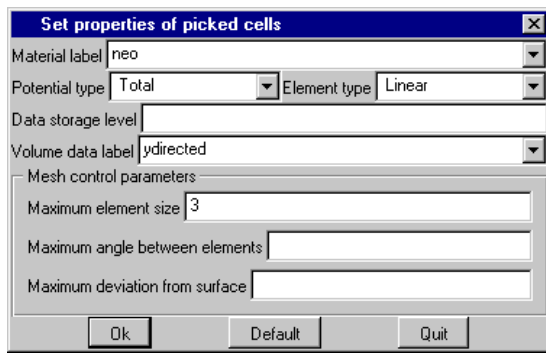
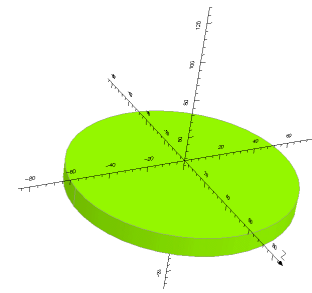
For this example, the magnet will be given the material name **neo** and the potential type will be changed to **TOTAL**. To do this, the volume must be selected as a **Cell** (for a discussion on **Cells** and **Bodies**, see [“Cells and Bodies” on page 5-1](#)).

Note that objects can be selected in order to hide them from the display (by first selecting the **Hide Entity** icon ) or to pick them for a subsequent operation

(using the **Pick Entity** icon ). See “Picking and Hiding Entities” on page 5-2 for more information.

<p>Ensure Pick Entity icon is selected.</p>  <p>Select Pick cell icon</p> 	<p>Move the mouse (without pressing the buttons) so that the cursor lies over the cylinder. The outline of the cylinder will be highlighted. Double click while the outline is highlighted and the solid cylinder becomes highlighted indicating that it has been picked.</p>	
------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-------------------------------------------------------------------------------------

The properties of the picked cell can then be set. From the top menubar, select

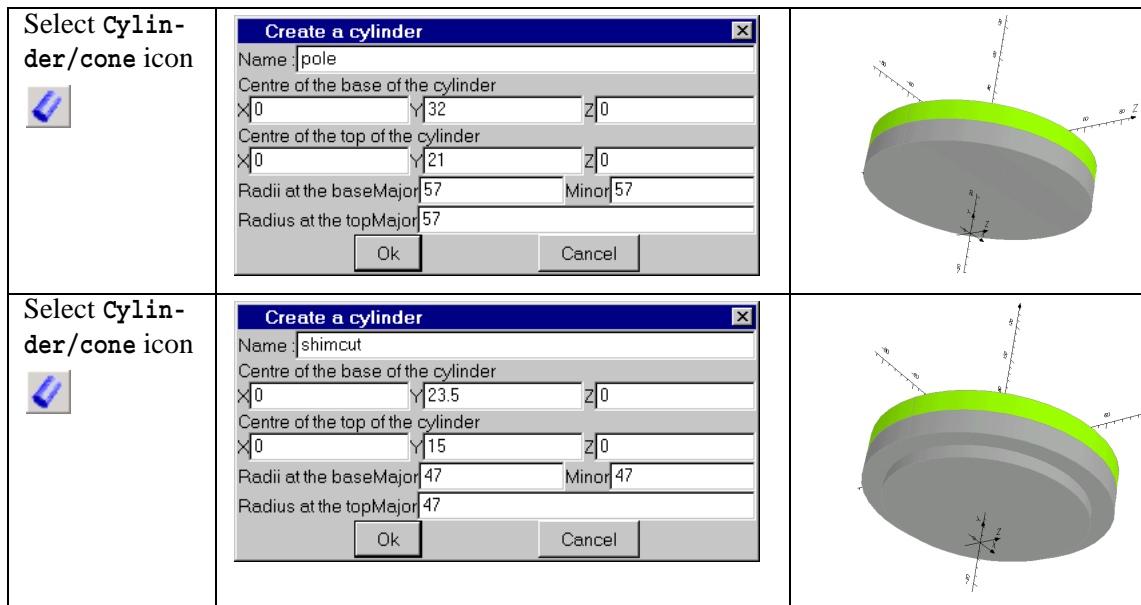
Properties → Cell properties		
		

Note that the potential type (**TOTAL**) can be selected from a menu by clicking on the arrow at the right side of the input box. The **Volume data label** is set to **ydirected**, will be used later to specify the easy direction of magnetisation for the magnet. The **Maximum element size** (3) specifies that no element in the magnet will be larger than 3 units along any of its edges. The magnet cylinder now becomes green, indicating it is made from the first new material name to be introduced. The material name **neo** is only a label (a logical name) - the physical properties of the material are yet to be defined.

Building the Pole and Shim

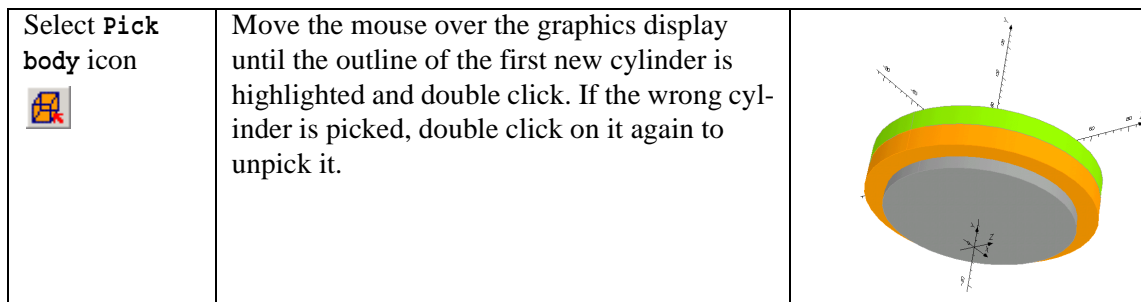
The pole and shim are constructed as a single volume of the model. This is achieved by initially making a cylinder as deep as both the pole and shim and then cutting a 2.5 cm recess using a Boolean subtraction of a second cylinder from the


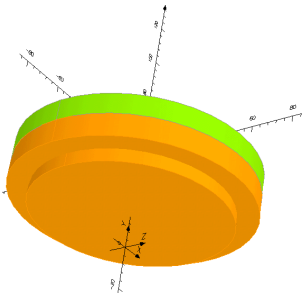
initial cylinder. Note that the cutting cylinder is deliberately defined longer than the depth of the recess. This makes picking the two volumes simpler.



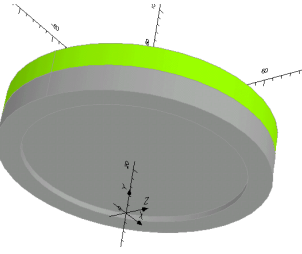
Boolean Subtraction

Boolean operations (union, subtraction, intersection, repair and trimming) are performed on bodies, not cells. The main distinction between a cell and a body is that a cell is a volume of the model, while a body is a hierarchical assembly of cells, faces, edges and points. Hence, it is necessary to pick the new cylinders created as bodies not cells.


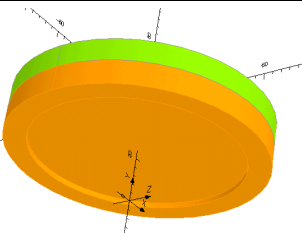
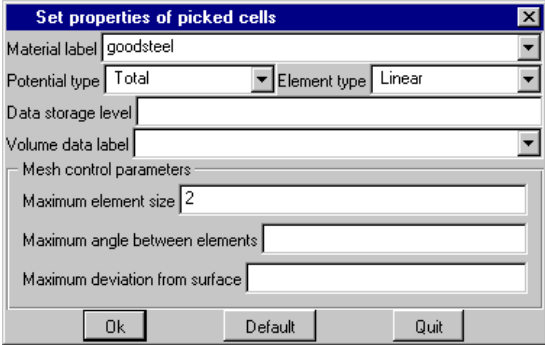
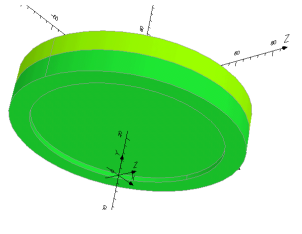


<p>Select Pick body icon</p> 	<p>Pick the second new cylinder in the same way.</p>	
-----------------------------------------------------------------------------------------------------------------------	------------------------------------------------------	-------------------------------------------------------------------------------------

Subtraction is performed from the **Operations** menu on the top menubar.

Operations → Combine bodies → Subtraction, with regularisation		
	<p>Subtract the body picked second from the first body picked. The second body disappears and the result leaves a single volume that is the result of the Boolean operation. Regularisation removes any internal faces that may be left as a result of the operation.</p>	

The properties of the combined pole and shim volume can now be defined from the **Properties** menu.

<p>Select Pick cells icon</p> 	<p>Highlight the new volume.</p>	
Properties → Cell properties		
		

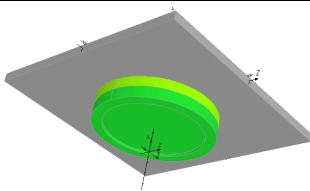
Building the Frame

The frame is constructed from two blocks - one to make the back plate and the second to make one of the (half) legs. The leg block is then copied to make four (half) legs. The back plate will be constructed by using keyboard entry. A block is a cuboid and may be defined using any two diagonally opposite corners


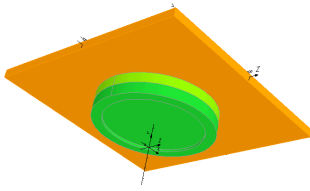
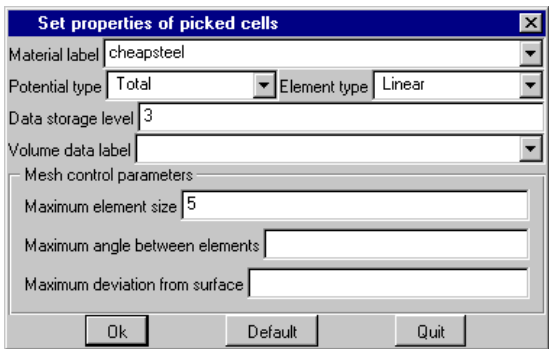
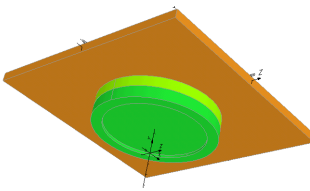
Using the Keyboard Entry Console

The keyboard entry console is activated from the **Windows** top level menu (as discussed previously) and by typing at the **OPERA-3d>** prompt. There are no spaces between the parameter name, the equals sign and the value to which the parameter is assigned.

A block is defined using the following commands:

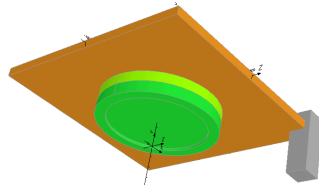
Windows → View console		
	<pre>OPERA-3d> block name=plate, OPERA-3d> x0=-100 y0=42 z0=100, OPERA-3d> x1=100 y1=50 z1=-100</pre>	

The cell properties can then be defined.

Select Pick cells icon 	Highlight the block.	
Properties → Cell properties		
		

The **Data storage level** is set higher than the default (1). When the plate and the legs are combined later to make a single volume using a Boolean union operation, the cell with the highest data storage level defines the properties that the combined body retains.

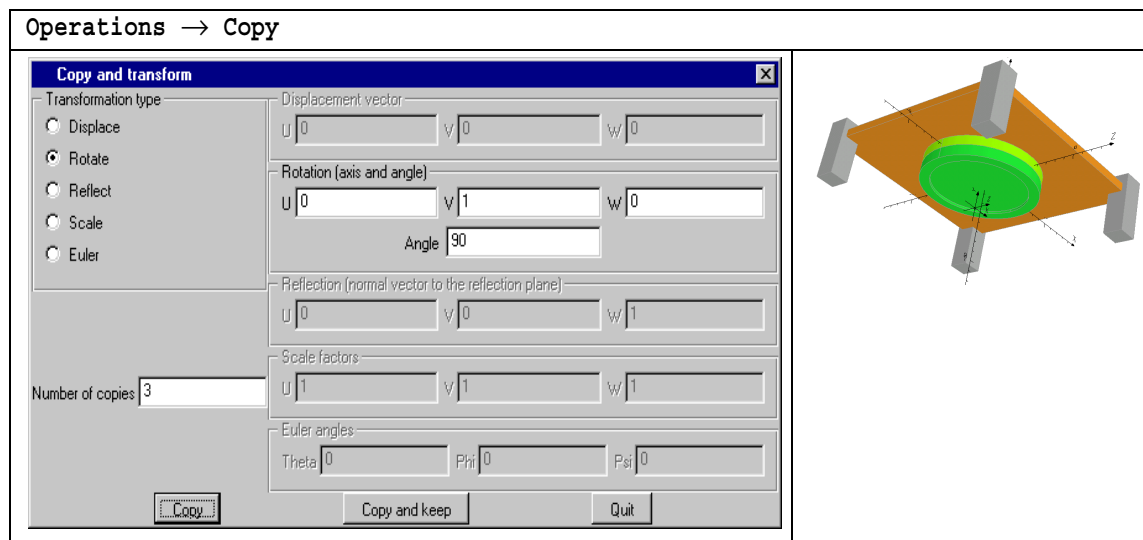
The first (half) leg is also defined using the keyboard entry. The parameter names are not included and the values will be assigned to the parameters in the defined order shown in the reference manual.

Windows → View console		
	<pre>OPERA-3d> block leg 85 0 85 105, OPERA-3d> y1=60 105</pre>	

Making Copies


The additional three copies of the leg will be made by rotation around the Y-axis. Copying is an **Operation** that can only be performed on a body. The left side of the copying menu defines the operation that is to be performed. Only the values on the right side of the menu that match the operation selected may be entered. The number of copies at the bottom left defines how many additional bodies will be constructed (not including the body that was originally picked).

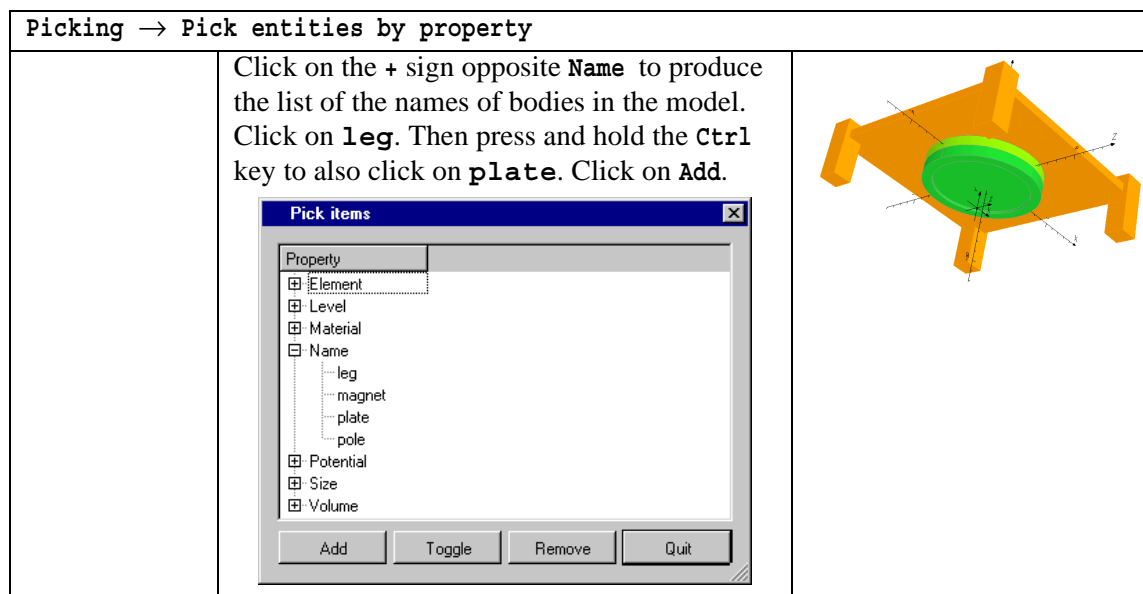
Select Pick body icon 	Pick the leg	
------------------------------------------------------------------------------------------------------------------------	--------------	---------------------------------------------------------------------------------------

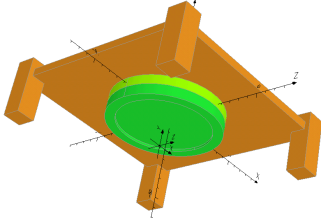


Picking Entities Using Their Properties

The back plate and the legs are combined into a single volume using a Boolean union operation. Bodies may be selected by one of their attributes using the **Pick-**

ing menu or the **Pick by property**  icon.



Operations → Combine bodies → Union, with regularisation		
	<p>Note that the legs turn to the same colour as the back plate, indicating they have used the data from the plate cell as it had a higher data storage level. Other results from using REGULARISATION are discussed in more detail in the next chapter.</p>	

This completes the geometry of the model.

Creating a Background Volume

The previous steps have defined the geometry of the magnet. However, the magnetic field must be determined in “all space”. The free space that surrounds the magnet must also be included as part of the model. In the real world, this space extends to infinity. However, in a finite element model the free space region is terminated at a distance from the region of interest, such that the termination does not significantly affect the accuracy. In this calculation, the free space region will be terminated 250 cm from the centre of the magnet in the horizontal plane (X and Z directions) and 150 cm in the vertical (Y) direction. Note, for real MRI magnets the termination boundary should be at least 10 times greater than these distances.

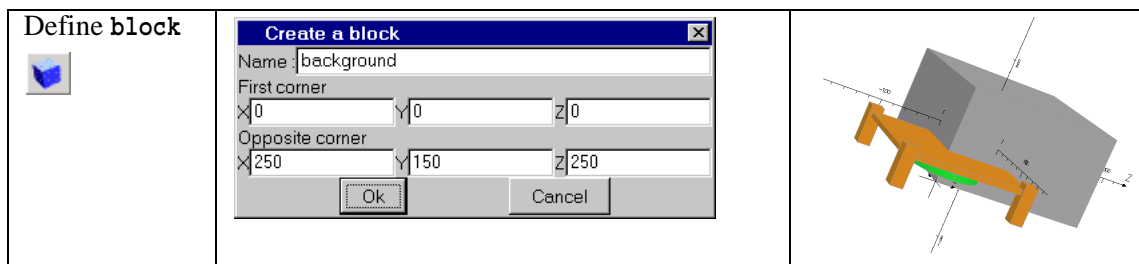
Exploiting Symmetry

Background Volumes

As explained at the beginning of this example, only 1/8 of the complete magnet needs to be solved because of symmetry. The 1/8 model will be created using a background volume, which will also define the outer boundary of the problem.


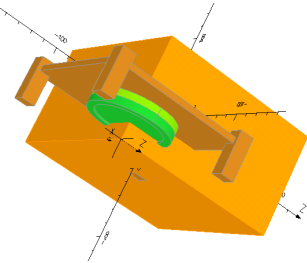
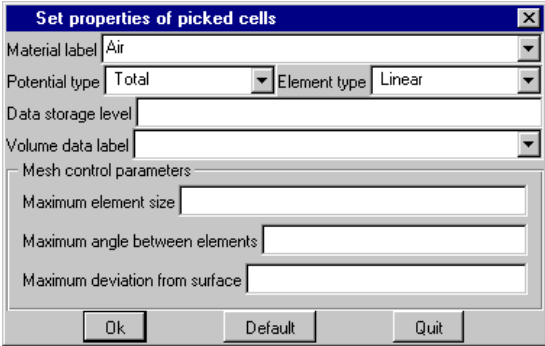
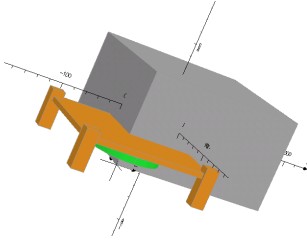

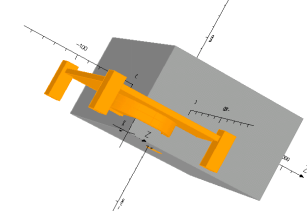
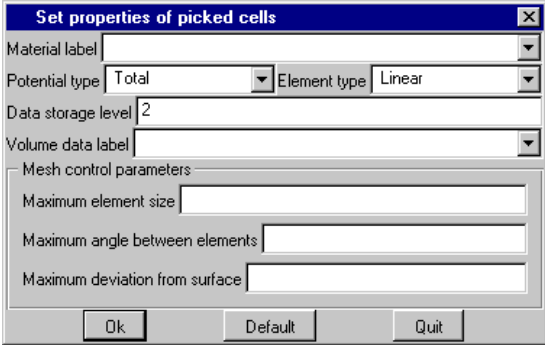
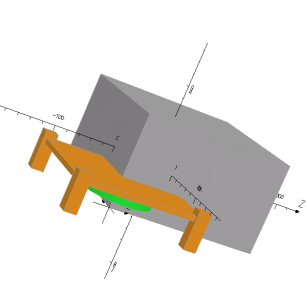
Each model to be analysed can only contain one background volume. Any part of the geometry that extends beyond the background volume is automatically removed for the analysis. Geometry that lies within the background volume takes precedence over the background i.e. the background volume only fills the parts of the space that are not already occupied, and additional geometry can be placed in the background after the background has been defined.

Background regions can be a volume of any shape. In this model, a block region is used. The background property is invoked by naming the volume **background**.




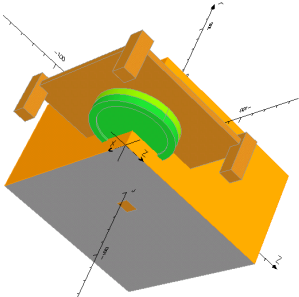
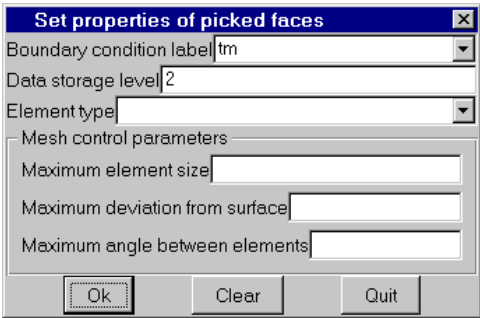
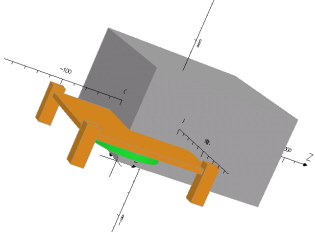
Setting the Background Properties


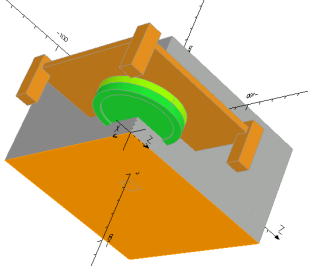
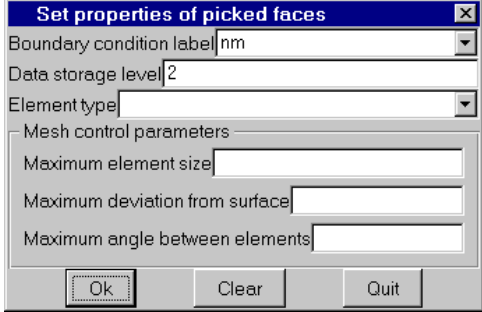
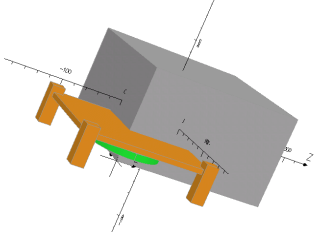
The properties of the background volume will only be invoked when a single body for the complete model is created. There are three data definition steps to be completed first. The first two set the **Cell properties** of the background regions and the MRI geometry.


Select Pick cells icon 	Pick the background region.	
Properties → Cell properties		
		
Select Pick cells icon 	Pick all the cells that make up the MRI geometry.	
Properties → Cell properties		
		

Boundary Conditions

The final data definition step before the model is created is to add boundary conditions to the background volume. Similarly to material properties, boundary conditions are applied by assigning a label (or logical name) to a face of the model. The label is then associated with the real boundary condition to be applied. This is especially useful if the same model has to be analysed with many different boundary conditions e.g. in electrostatics, as the surfaces for the new condition do not have to be reselected - only the boundary condition is modified.

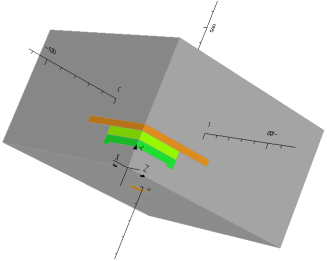
<p>Select Pick faces icon</p> 	<p>Pick the XY and YZ planes of the background volume. The outlines of the faces will be highlighted to indicate which face will be picked. On these two planes the magnetic field will be entirely tangential.</p>	
<p>Properties → Face properties</p>		
	<p>The label, tm, chosen reflects that a TANGENTIAL MAGNETIC boundary condition will be applied later. However, this is only a label and any other character string could also be used. The higher data storage level ensures that the boundary conditions from the background volume will be transferred to all new faces created within the area of the background faces when the symmetry is imposed.</p> 	

<p>Select Pick faces icon</p> 	<p>Pick the ZX plane on the background volume.</p>	
<p>Properties → Face properties</p>		
		

The model can be saved in a file at this time by selecting the **Save**  icon or typing **Ctrl-S** (hold down the **Ctrl** key and type **S**). Any name can be given in the browser window that follows. The model can be subsequently saved at any other time without respecifying the file name.



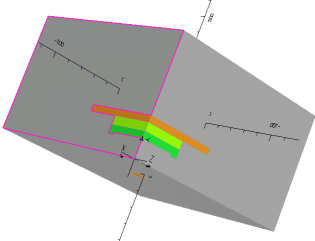
Making the Single Body Model

Before the finite element mesh can be generated, the model must be made into a single body.

Model → Create model body		
	<p>The properties of the background volume have taken effect with only the 1/8 model now being displayed. The geometry of the complete half MRI is still held in the model but will not be used. Selecting Delete model body from the Model menu will undo the last step, if the user wishes to verify this!</p>	

The higher data storage level on the geometry of the MRI has ensured that the faces of the MRI geometry are displayed on the symmetry planes rather than the faces of the background volume.

However, the faces of the MRI geometry have inherited the boundary condition label applied to the background volume. This can be verified as follows.



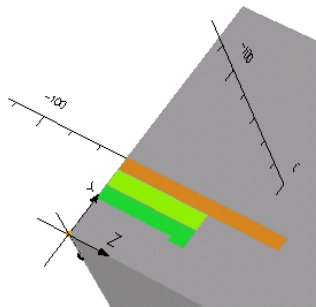

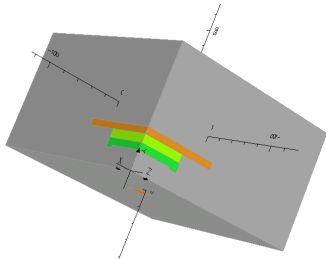
<p>Select Pick faces icon</p>  <p>Select Show entity properties icon</p> 	<p>Double click on any face on the symmetry planes. Note that the faces of the background volume on the symmetry plane are no longer simple rectangles or squares, but fit round the MRI geometry.</p>	
---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	---------------------------------------------------------------------------------------

Building the Finite Element Mesh

The quality of the finite element mesh in the good field region between the poles is very important for accurate modelling of MRI magnets. The volumes making up the pole / shim, magnet and frame of the MRI have already been given a mesh control size of 2, 3 and 5 respectively, but no mesh control size was applied to the background volume. When the mesh is constructed a global mesh control size parameter is given which is applied to all volumes in which a mesh control size is not given.

Controlling the Mesh Size Using Geometric Entities

It is possible to use other geometric entities in the model - faces, edges and vertices - to set the mesh control size around particular features. In this model, the mesh at the origin will be controlled to provide accurate modelling in the good field region.

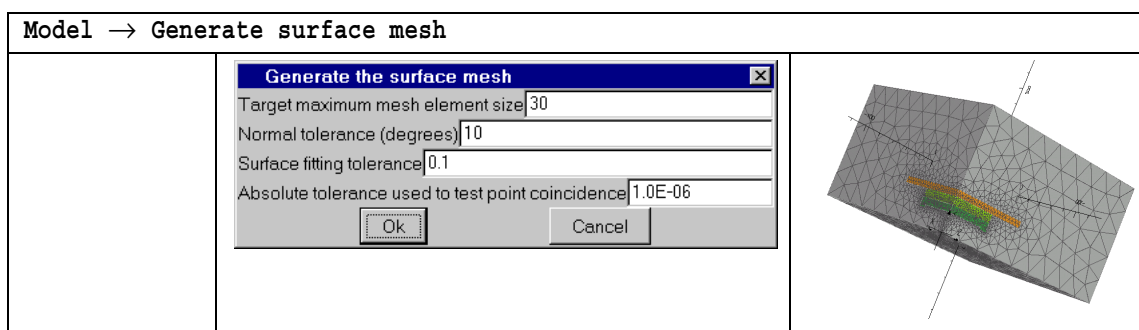
<p>Select Pick entity icon</p>  <p>Select Pick vertices icon</p> 	<p>Move the mouse over the model until the vertex at the origin is highlighted. Select the vertex with a double click.</p>	
<p>Properties → Vertex properties</p>		
		

A warning message is issued that this change will be lost if the user returns to **COMPONENT** mode. However, sometimes some properties have to be entered in **MODEL** mode as the entity did not exist before the background region imposed the symmetry.


The model is now ready to be meshed.

Surface Meshing

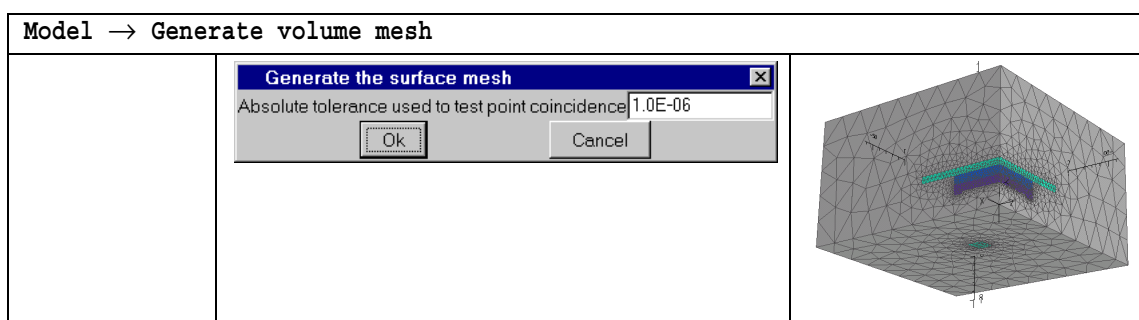
Creation of the finite element mesh is performed in two stages. All surfaces of the model are initially meshed with triangles. This includes internal surfaces between volumes of the model. After the surface mesh is complete, the volume mesh can be created.



Note the concentration of surface elements in the region of the origin and the effect of the mesh control size values set in the MRI cells. This process may take one or two minutes dependent on processor speed etc. If the user starts the surface (or volume) meshing with incorrect parameters, the process may be aborted by

hitting the Cancel icon  - a very useful tool! As each **FACE** is meshed it is highlighted by its outline. Progress can also be monitored in the status bar at the bottom.

Volume Meshing



Construction of the volume mesh takes a few minutes and results in a mesh with just over 150,000 elements and a little more than 25,000 nodes - the exact numbers will depend on the computer hardware in use. Each **CELL** is also highlighted as it is completed.

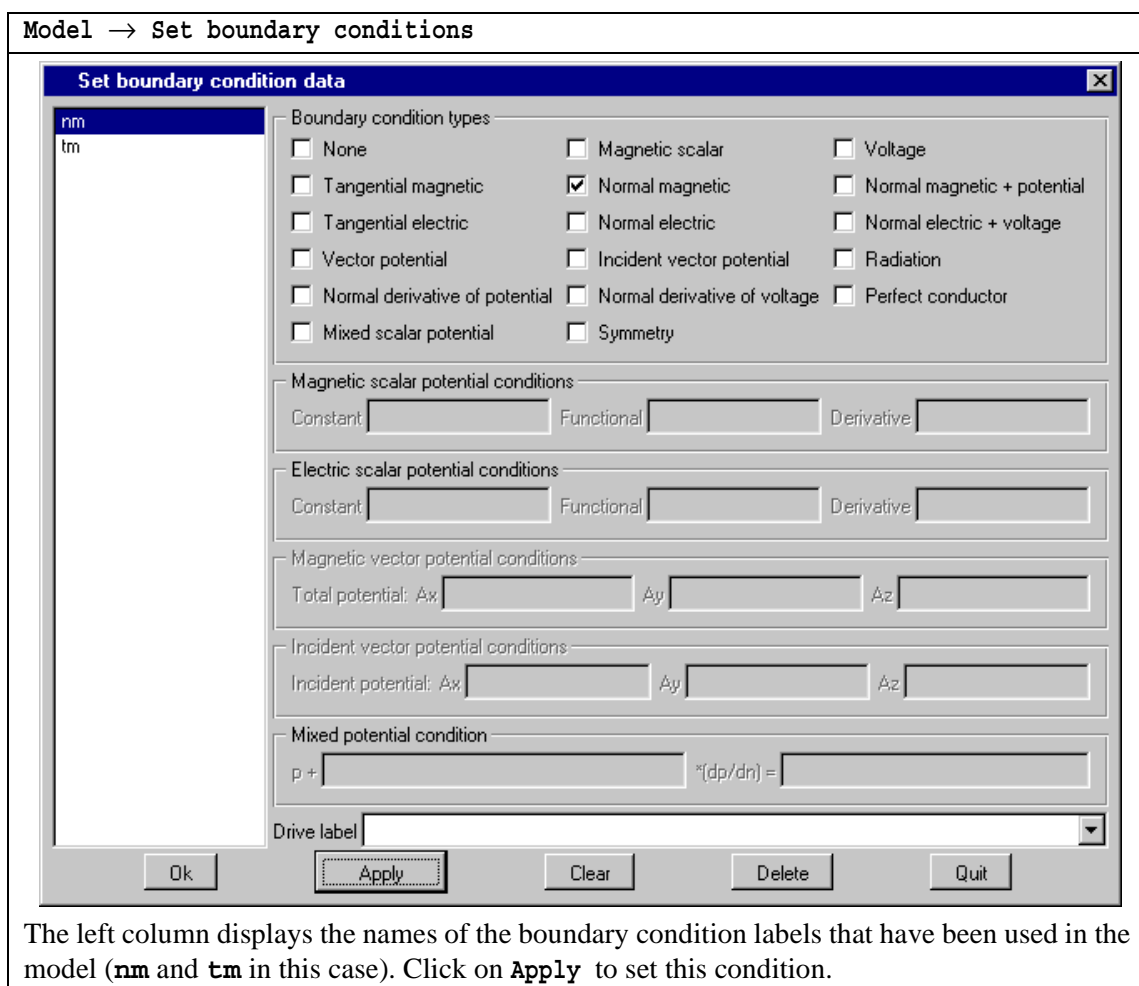
Preparing the Model for Analysis

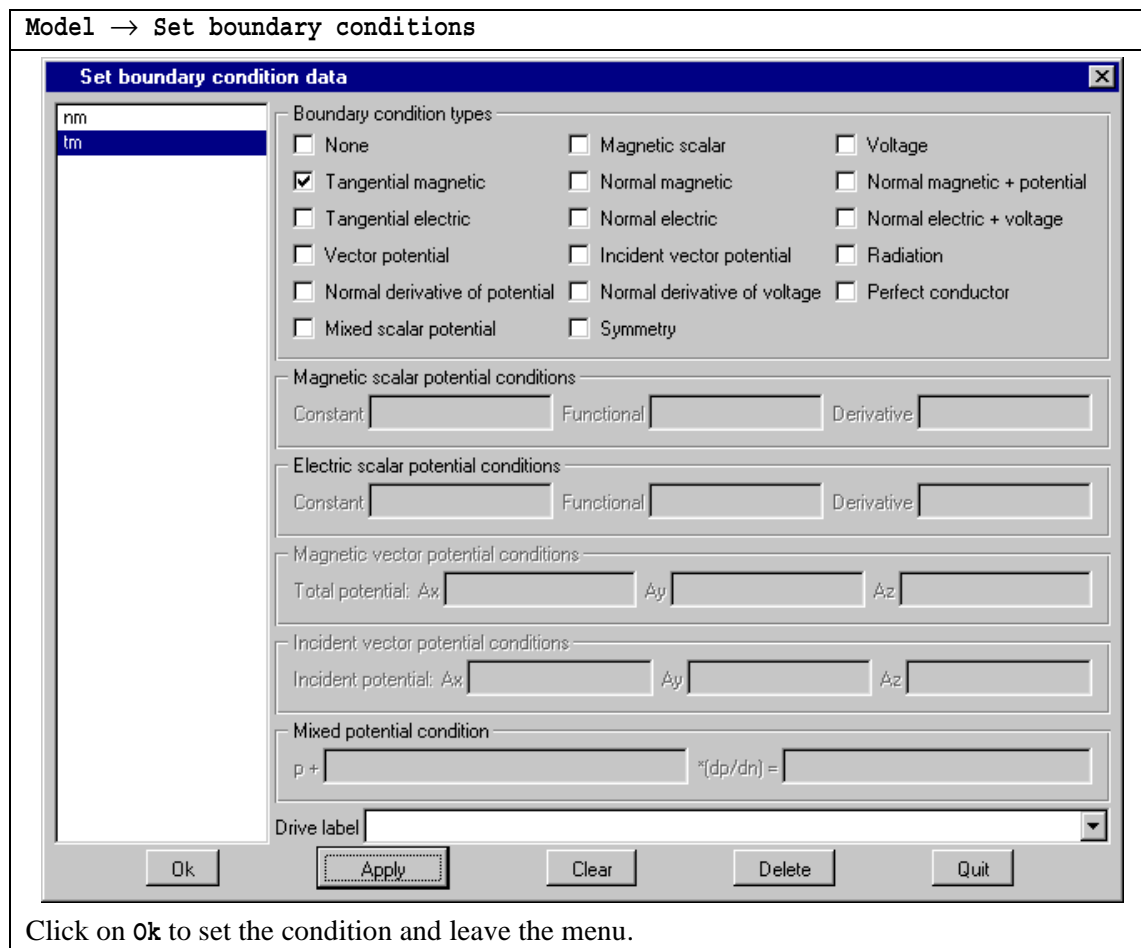
Analysis Data

The model is now ready to be analysed. Firstly, the physical meaning of the boundary condition labels, the actual material properties and the easy direction of the permanent magnet are specified.

Setting Boundary Conditions

Starting with the boundary condition labels.





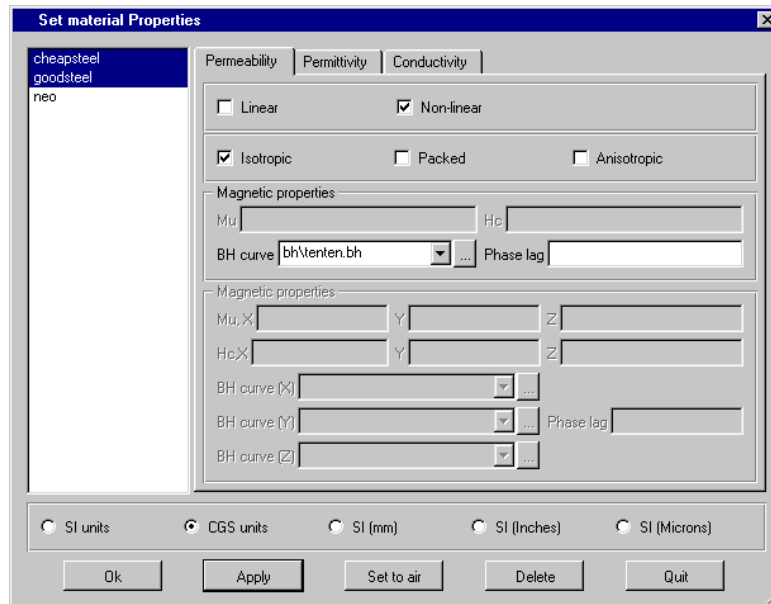
Click on **Ok** to set the condition and leave the menu.

Setting Material Properties

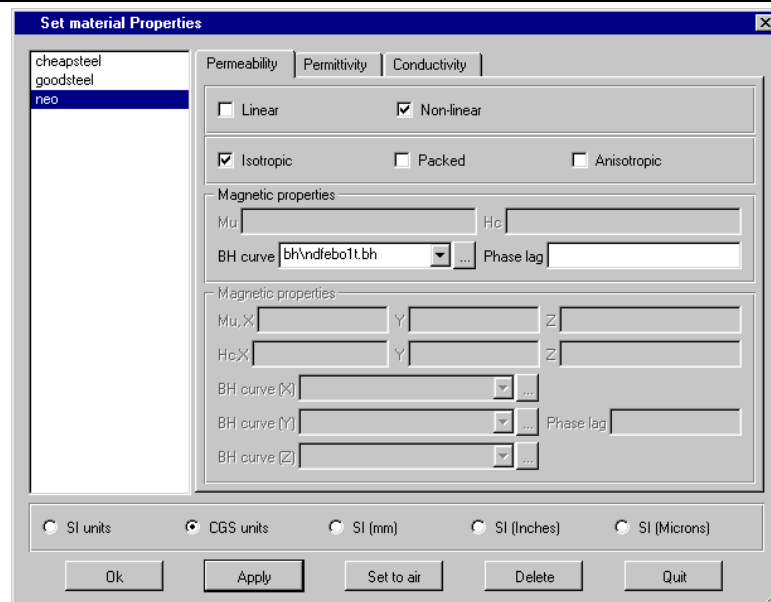
Material properties are defined in a similar way to boundary conditions. In this analysis, both **goodsteel** and **cheapsteel** will be given the same magnetic characteristic. The permeability of the magnetic materials will be defined using a non-linear characteristic curve. If the analysis is run linearly, the initial slope of the curve will be used. If the analysis is run non-linearly, the curve will be used to match the permeability of each element to the flux density by iterating. The materials will be treated as isotropic.

Clicking the button on the right beside the **BH curve** entry in the **Isotropic properties** section allows the user to browse through the computer file system to find the BH curves. For this analysis, curves that have been supplied with OPERA will be used. A sub-directory, *bh*, exists in the installation directory for OPERA.

Model → Set material properties



Double click on the *tenen.bh* file name - a curve for ANSI 1010 steel - to select this for the curve for both **goodsteel** and **cheapsteel**. Click on **Apply** to use the selections.

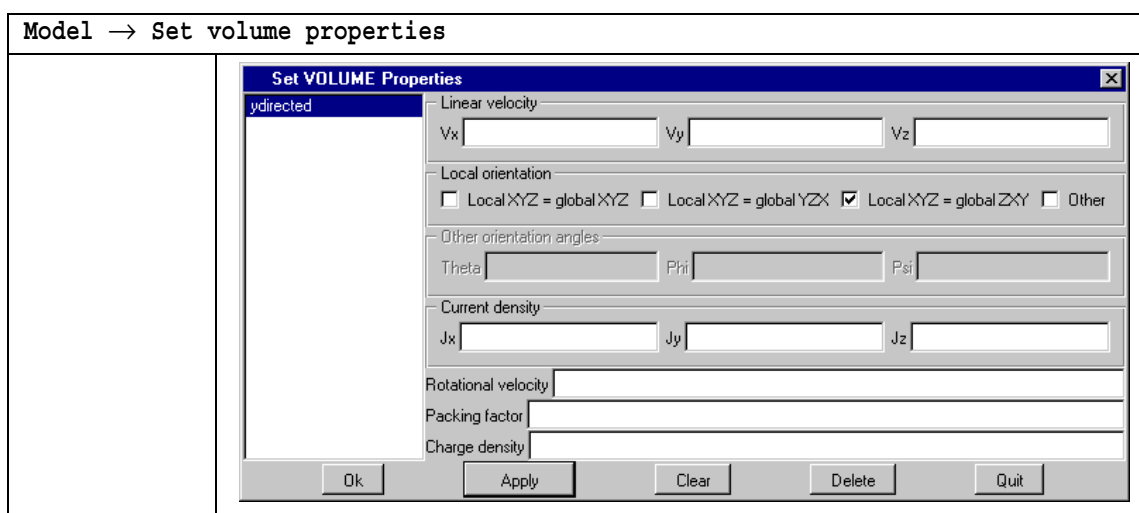


Select the *ndfebo1t.bh* curve from the same directory as the *tenen.bh* curve. Click on **Ok** to use the selections and leave the dialog.

Setting Volume Properties

The final action before writing the data for analysis is to specify the magnetisation direction for the permanent magnet. The default direction is along global Z, but in this model the magnet is magnetised through its thickness i.e. in the global Y direction. Each cell of the model may have an assigned **volume label**. These labels are used to associate a set of physical properties - velocity, charge density etc. - with a particular volume or volumes of the mesh.

The Z direction of a local coordinate system in the volume specifies the magnetisation direction. The orientation of the local coordinate system is controlled by Euler angles termed **Theta**, **Phi** and **Psi** on the menu. There are also boxes that can be checked to access the most commonly used systems.



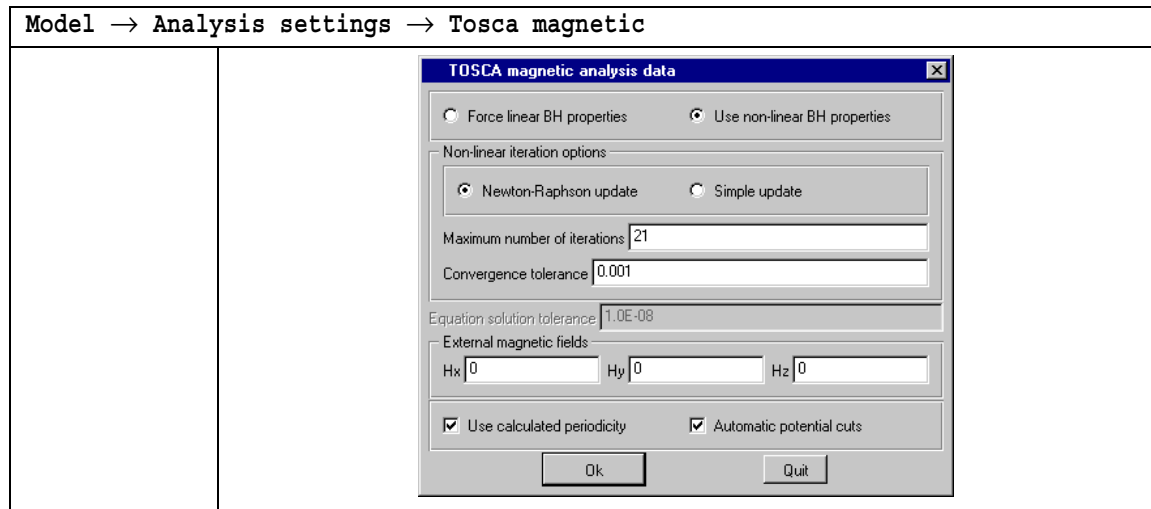
Note: This is equivalent to setting **Theta=90**, **Phi=90**, **Psi=180**. Click on **Ok** to specify the orientation and leave the dialog.

Creating the Analysis Database

There are two steps to creating the analysis database. Firstly, the user sets appropriate analysis information for the type of analysis program used to solve the field equations. The second step is to create the database itself.

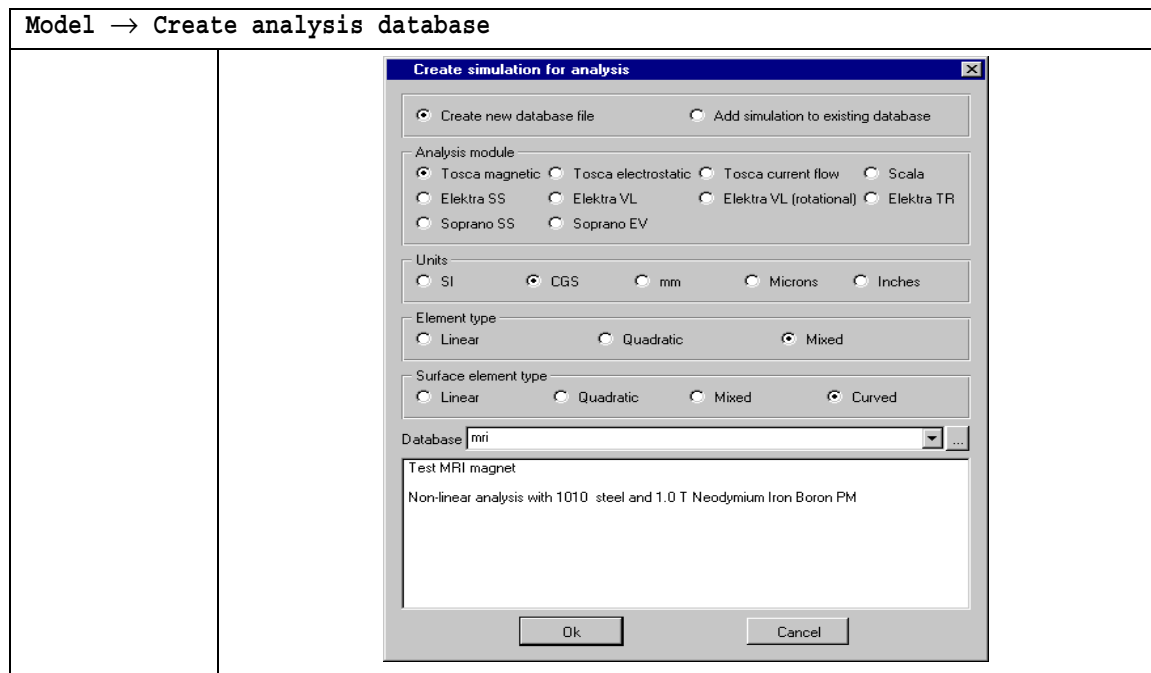
Analysis Settings

The dialog allows the user to choose a linear or non-linear analysis, the number of non-linear iterations, the type of non-linear algorithm used to converge the non-linear solution and other similar information.



Analysis Database

The final step is to create the analysis database. Database files are given the extension *.op3* automatically. Browsing is available by clicking on the button at the right of the filename box, or previously used names can be recovered using the down arrow, also on the right. The units in which the model has been constructed are selected at this time. In the large box at the bottom, the user may enter a multi-line title to describe the contents of the database.



Close the following message box after checking that the data has been defined correctly.

Saving the Model

The model is stored as a *.opc* file. The model can be stored at any time during the process of creation (as shown on [page 4-18](#)) and can be read in again to allow additions or modifications. The user may re-save the model using the name previously chosen or select a new name as follows.

File → Save as new model data		
	Enter the name <i>mri</i> in the name box. The file extension will be automatically supplied to create the file <i>mri.opc</i> .	

The model file created will not only include the geometry, attached labels, mesh control parameters and data level assignments, but will also retain the boundary conditions, material and volume properties, and the analysis settings for all the types of analysis specified by the user. It is also possible to save the model with its mesh in binary format, using **Save model with mesh** which is also found on the **File** menu.

File → Exit		
	Finish the Modeller.	

The data file is also provided with the OPERA installation, in the sub-folder *Examples/3D*, if required.

Running the Analysis

Microsoft Windows Platforms

After exiting the Modeller, the user accesses the OPERA Console window again. From the **OPERA-3d** menu, select **Interactive Solution**. This pops up a dialog from which the user can select the analysis program to be run (TOSCA) and use the browser to select the file *mri.op3*. Click on **OK** to start the analysis. A new window appears showing the progress of the analysis.

Unix Platforms

After exiting the Modeller, control returns to the OPERA window. The user selects the analysis type (TOSCA) and gives the name of the file to be analysed, *mri*, without the *.op3* extension. For an interactive solution, choose option **n** (for now). The analysis starts immediately in an OPERA window.

Running Analysis from the Modeller

The Modeller also allows the analysis to be run interactively from **Start analysis** on the **Model** menu. However, this requires sufficient memory (real and virtual) to allow both the Modeller and the analysis program (TOSCA) to be active simultaneously. Consequently, for this first example, solution from the OPERA Console or window is chosen to minimise memory requirements to complete a successful OPERA session.

Monitoring Progress

The program reports when major stages of the analysis are complete and the amount of CPU and elapsed time used at each stage. To obtain a non-linear solution, an iterative method is used. Consequently, the same stages are repeated until convergence of the solution is obtained. The following text shows the processes that are performed for one particular iteration.

```

Checking Nonlinear progress      : 60.3 s cp, 1.0 m elapsed

L2 norm of residual= 0.28201E+06

Calculating non-linear matrix      : 36.2 s cp, 0.6 m elapsed
Pre-conditioning matrix problem    : 0.3 s cp, 0.0 m elapsed

DSLUCG iterations: 185 Relative change= 0.96030E-10

Iterative matrix solution          : 19.7 s cp, 0.3 m elapsed

Biggest residual = 31041.2862172744 for equation 14829

Newton:Relaxation= 1.000000000000000 Residual= 113888.128111167

Jacobean derivative relaxation factor = 0.957E+00

Maximum change at node 9797 (equation 9797)

Coordinates of node are 56.7553538538332 40.8781981722246 5.27539656577354 (cm.)

RMS change in solution= 0.430E-01 Maximum change= 0.398E+02 Relaxation= 0.100E+01

ITERATION 4 complete on 23/Nov/2001 at 16:02:23

```

The program prints out the amount of time the previous iteration took to see if the solution had converged and forms a new matrix of equations to solve by updating the permeability distribution from the previous solution. It then solves the new equations and determines the over (or under) relaxation factor that will minimise the L2 norm of the residual. Finally, it determines both the relative RMS and maximum changes of the solution from the previous solution to see if the solution has converged.

The convergence criterion is that the maximum change divided by the relaxation factor should be less than the user defined convergence tolerance. This model should converge in 9 iterations, and takes about 530 CPU seconds on a 400 MHz Pentium processor PC. A summary message of the total CPU and elapsed time is printed at the end of the analysis, after which post processing can begin.

Post Processing

The post processing shown here is typical of the major requirements for an MRI system. Many other features are available in the post processor and are covered in other worked examples in the OPERA-3d User Guide.

Microsoft Windows Platforms

From the **OPERA** Console window, select **Post-Processor** from the **OPERA-3d** menu.

Unix Platforms

The **OPERA** window offers a choice of **OPERA-3d** modules again. The user enters **post**.

Loading the Solution

When the post processor starts, the display shown in Figure 4.5 is obtained.

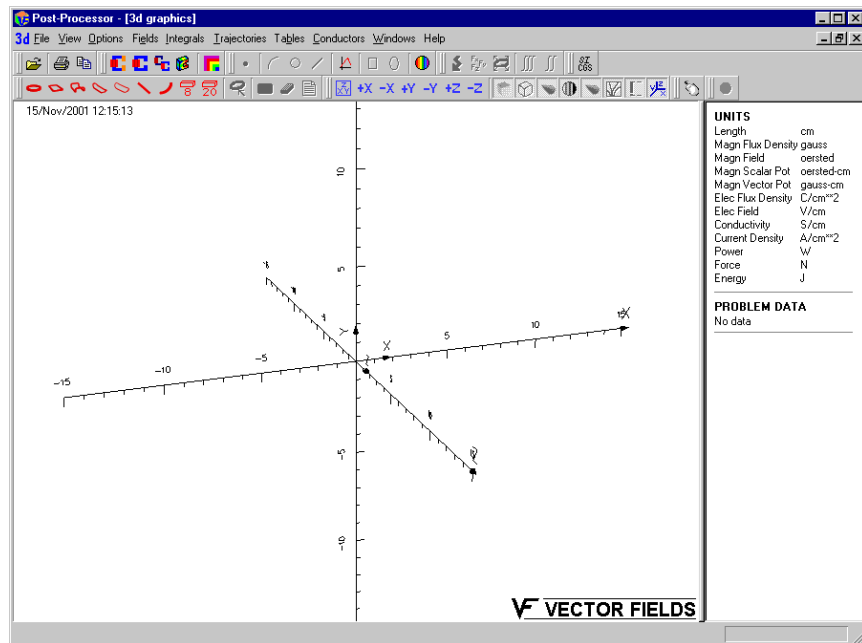
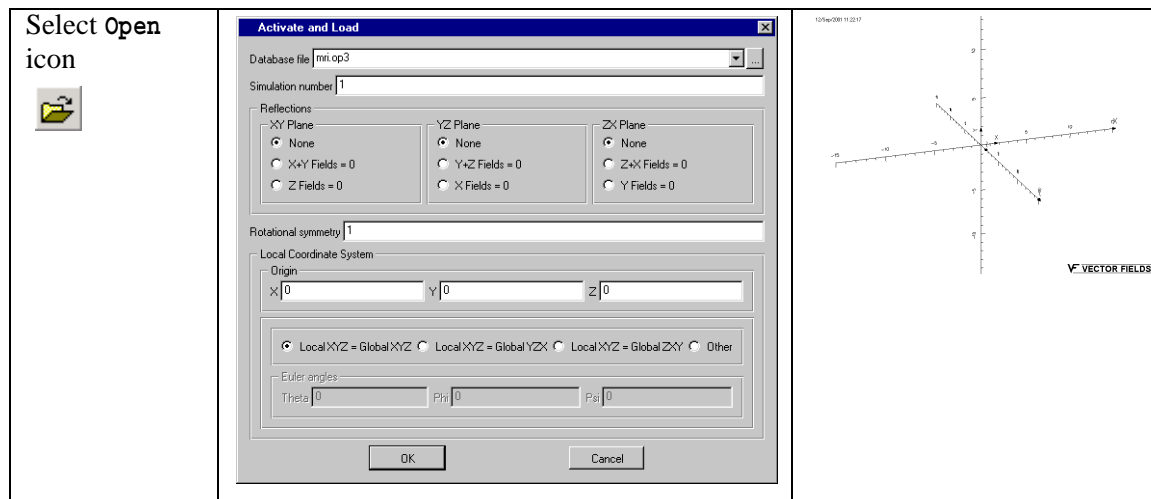


Figure 4.5 Initial OPERA-3d post processor window


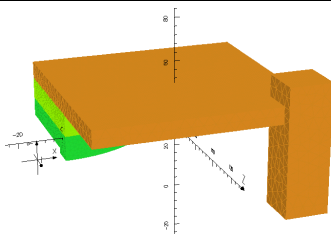
The post processing features are accessed from the menubar or from icons in a similar way to the Modeller. Many of the icons are similar to the Modeller and perform the same function. Keyboard access is also obtained in the same way.

Load the solution file as follows:



Select the database, *mri.op3* using the browser. Double clicking on the filename or clicking on **Open** in the browser enters the file name in the **Database File** box. Click on **OK** to load the file. Note that a message is given in the lower left corner of the window to say that a TOSCA magnetostatic analysis has been loaded. Details of the analysis data are given on the right side of the graphics area. However, the display area remains unchanged.

Displaying the Model

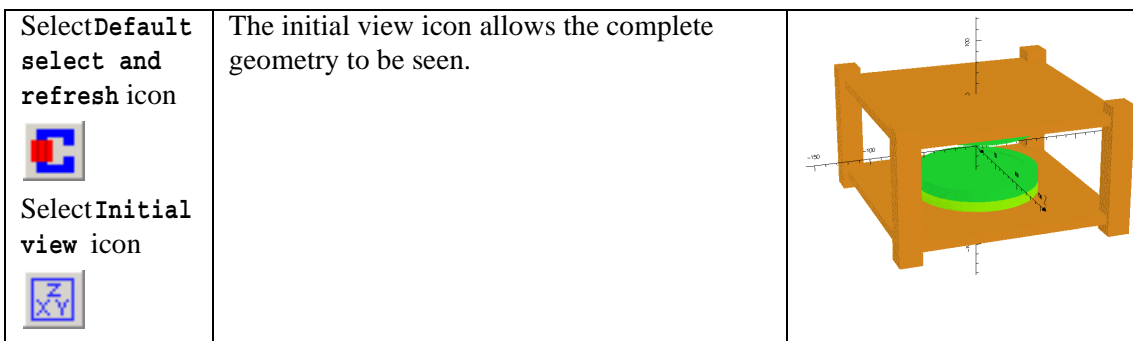
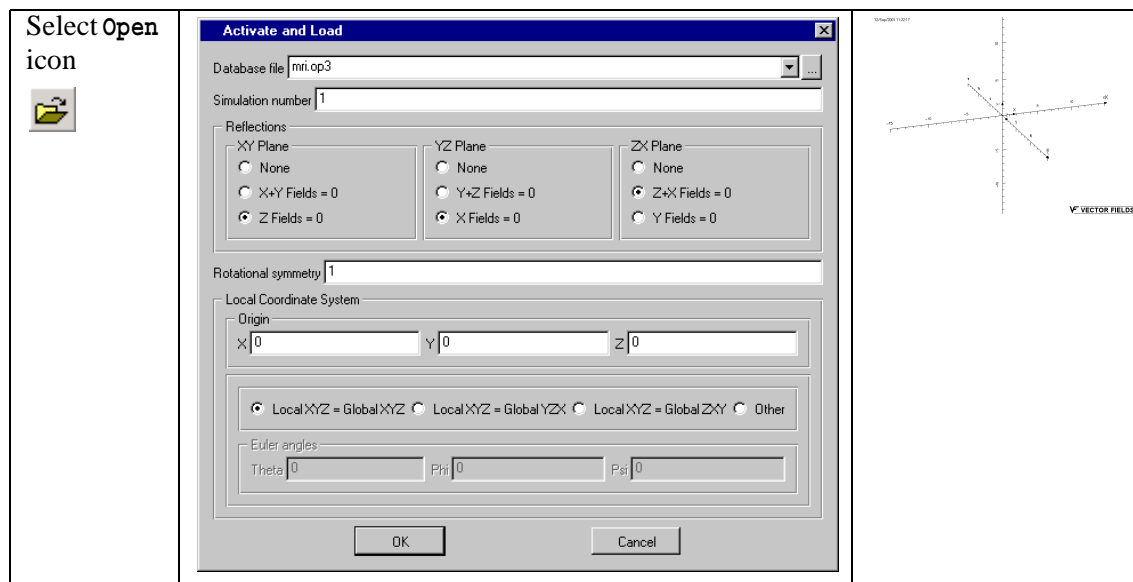
<p>SelectDefault select and refresh icon</p> 	<p>This will select all materials not called air for display by default. Note that the names of the materials used in the Modeller - goods-teel, cheapsteel and neo - are selected. The operation of the 3d-Viewer in the post processor has the same functionality as in the Modeller. A timer bar shows how the selection and display process is proceeding.</p>	
----------------------------------------------------------------------------------------------------------------------------------------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-------------------------------------------------------------------------------------

Replicating the Model

For computational efficiency, only 1/8 of the MRI and the surrounding free space was solved in TOSCA. The post processor allows the user to replicate the model to obtain the complete magnet - which will be necessary to perform some of the post processing calculations.

The reflections not only indicate the copies of the solved geometry but how the magnetic field should be replicated in the symmetric copies. Clearly, they should match the boundary conditions defined in the Modeller for correct display of the results.


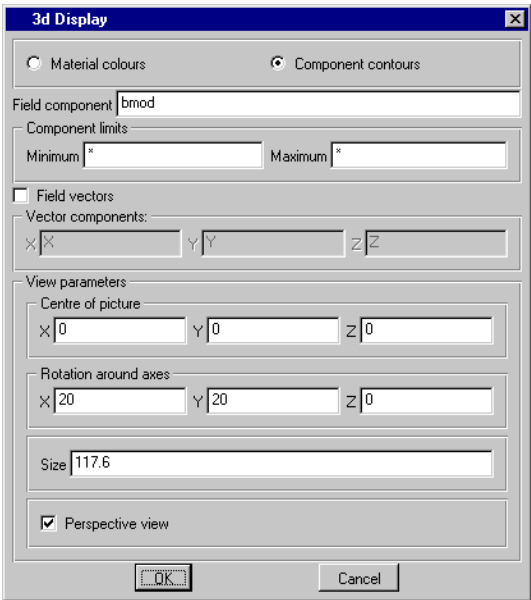
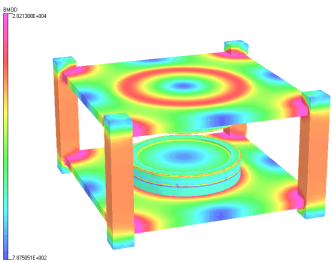


Use the radio buttons to create the full magnet. Then click on **OK**. Note that this removes the previous display of the geometry.



Checking the Solution

Results on the Geometry

Results can be superimposed on the geometry. Component BMOD gives the magnitude of the magnetic flux density

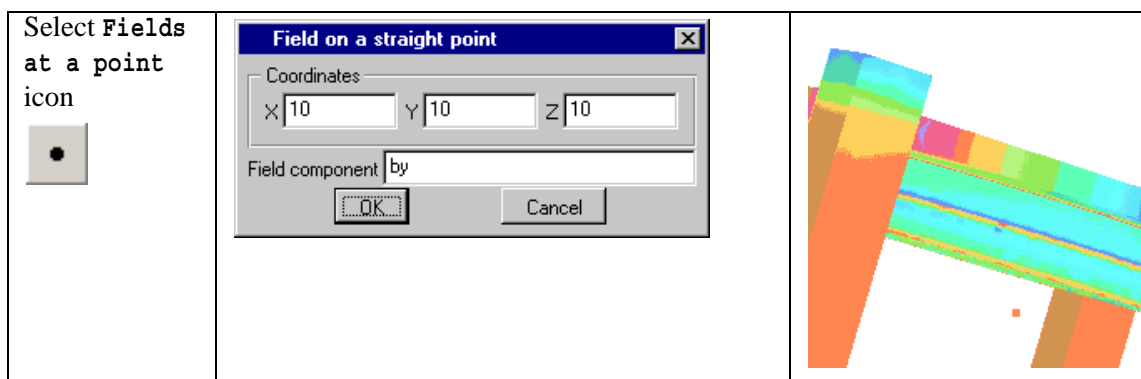
<p>Select 3D display icon</p> 		
<p>Toggle Outline view of model icon</p>  <p>and Axes icon</p> 	<p>The surface finite element mesh is initially displayed over the geometry as well as the contours. The mesh can be toggled on and off, as can the coordinate system axes and contour labels.</p>	

The scale to the side shows the magnitude of the flux density in Gauss. The user should check that the flux density values are about as expected. The maximum value of the flux density should be about 20 kGauss and the minimum close to zero.

The user should also check that the direction of the field is correct - both in the original model and the images. This can be achieved using field vectors in the 3D display but for this model a simpler check can be made.

Fields at a Point

To interrogate the field at a specific point:



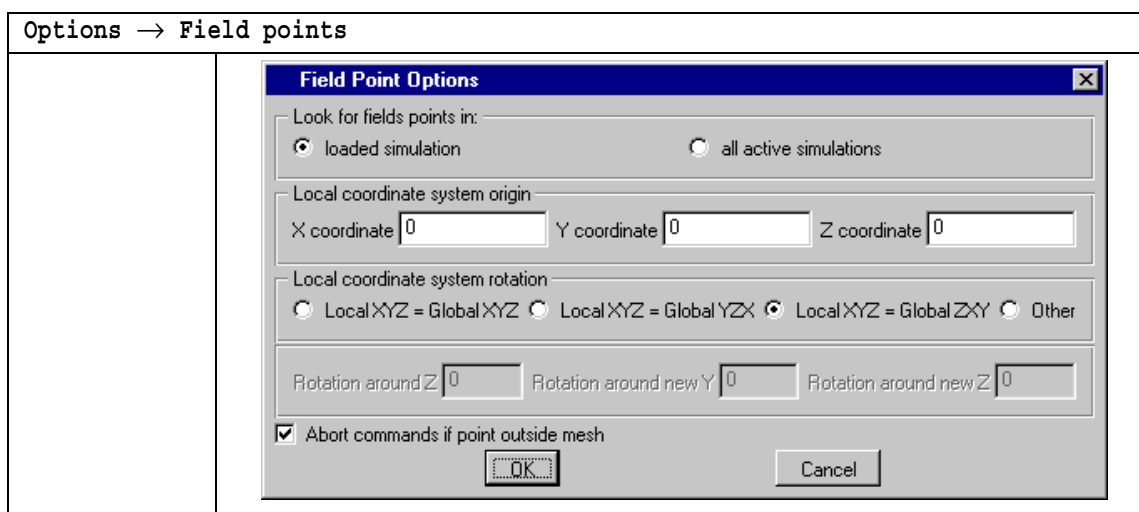
The value returned should be about +1870 Gauss. Note that the position of the point is highlighted. Because of the symmetry of the magnet, the Y-component of flux density values at $(\pm 10, \pm 10, \pm 10)$ should also be positive and the same magnitude. Check this by changing the coordinate values. If any of the values are negative, the reflections were not entered correctly when the complete magnet model was loaded.

Field Homogeneity

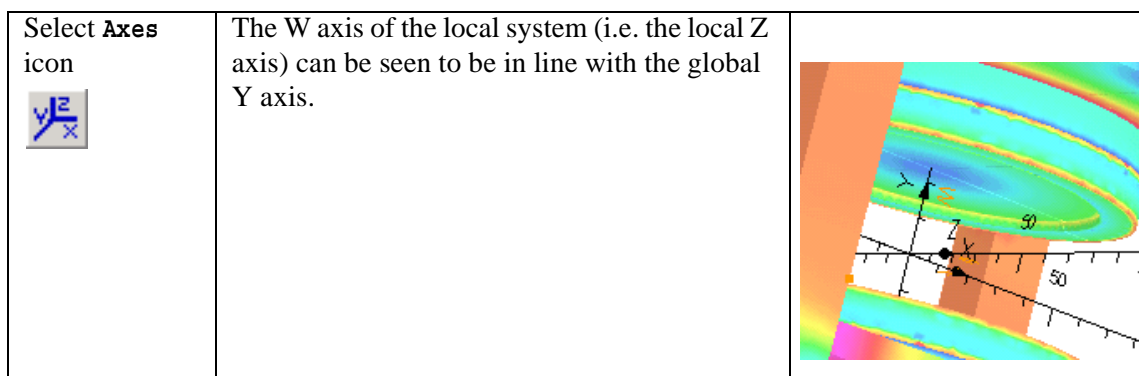
In MRI systems, it is important to have a volume of very homogeneous field. An effective way of representing the homogeneity is as the coefficients of associated Legendre polynomials.

Local Coordinate System

The post processor assumes that it will compute the coefficients using a spherical coordinate system, such that the field is acting in the direction of its Z-axis. In this magnet, the field is acting in the Y-direction. Consequently, the first task is to define a local coordinate system such that its Z-axis is along the global Y-axis.



The orientation of the local coordinate system can be seen by making the coordinate axes visible again.

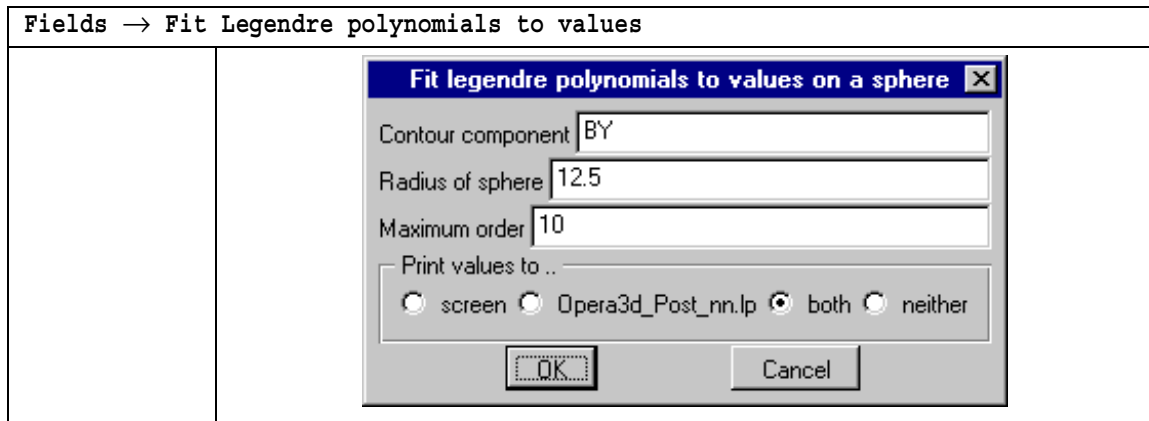


Legendre Polynomial Coefficients

The field component is still set at **BY** from the previous **Fields at a point** evaluation, which is the correct component. The fitting to obtain the Legendre polynomial coefficients is performed on a sphere with a defined radius. The program samples the field on the surface of the sphere and fits the results to the polynomial expansion. These are reported in a separate window. The form of the polynomial of order N is

$$B_y = \sum_{n=0}^N \sum_{m=0}^n P_n^m(\cos\Theta) [\alpha_n^m \cos(m\phi) + \beta_n^m \sin(m\phi)] \quad (4.1)$$

The table of coefficients shown are the α and β values in this equation. The order columns are n and m respectively. Consequently, the value of α for $n = m = 0$ is the mean value of the dipole field at this radius. If the sphere contains no sources (magnetic materials or currents), the associated Legendre polynomial is the solution to Laplace's equation and this value will be the field at the centre of the sphere.




Providing the field satisfies Laplace's equation, homogeneity on any other radii spheres can also be evaluated by inclusion of an additional r^n multiplier in the equation, where r is the radius normalised from the actual evaluation radius, i.e.

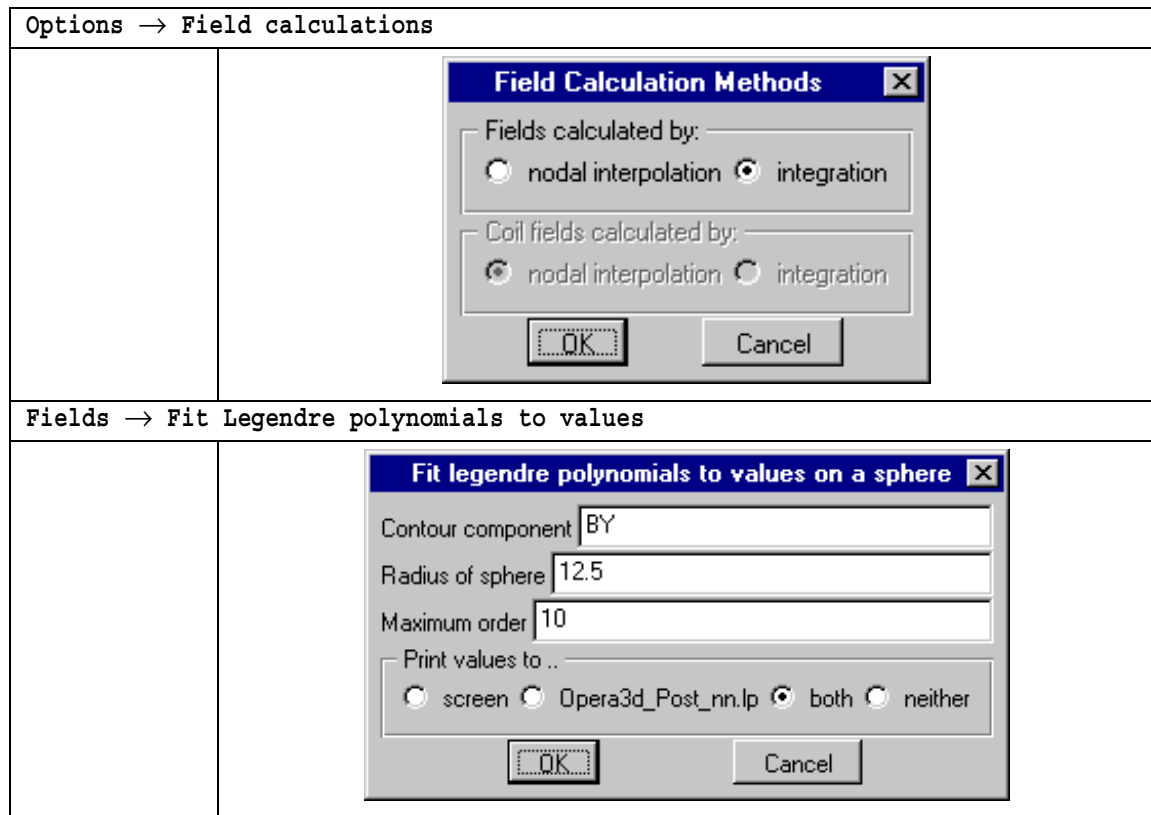
$$B_y = \sum_{n=0}^N r^n \sum_{m=0}^n P_n^m(\cos\Theta) [\alpha_n^m \cos(m\phi) + \beta_n^m \sin(m\phi)] \quad (4.2)$$

As may be expected from the symmetry of the magnet, the largest harmonic contaminants are at $n = 2, m = 0$ (-0.98 Gauss) and $n = 4, m = 0$ (0.26 Gauss). These probably result from the return flux through the legs of the magnet.

Integral Fields

In the above calculation, the values of field sampled on the spherical surface were computed using the element basis functions. The program determines which element the sampling point lies in and interpolates from the nodally averaged values. An alternative method of computing the field is to integrate the magnetisation of the magnet. This has the advantage that the value computed at the sample point is not dependent on the local mesh size. This is a fairly lengthy process, but for many problems it gives improved accuracy. Note that the process can be aborted

using the Cancel button .


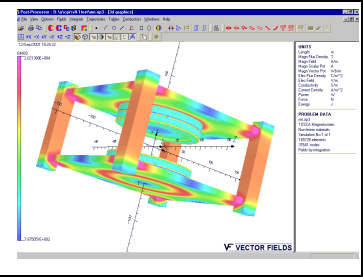


Force Between Poles

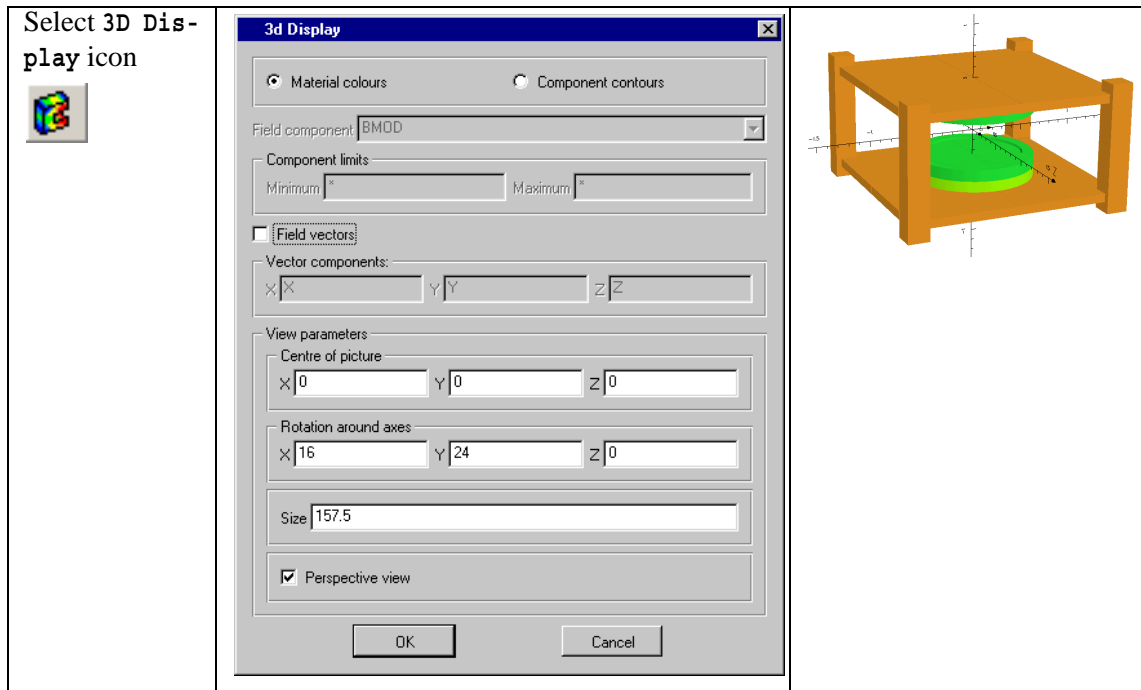
Generally, force calculations on magnetic objects are computed by integration of the full Maxwell stress tensor over a surface surrounding the object. For a symmetric magnet such as this one, it is sufficient to compute the integral of $\frac{(B_y)^2}{2\mu_0}$ on the mid-plane between the poles. This formula is valid for SI units.

Changing the Unit System

Even though the model was created in CGS units, it is possible to change the unit system in the post processor.

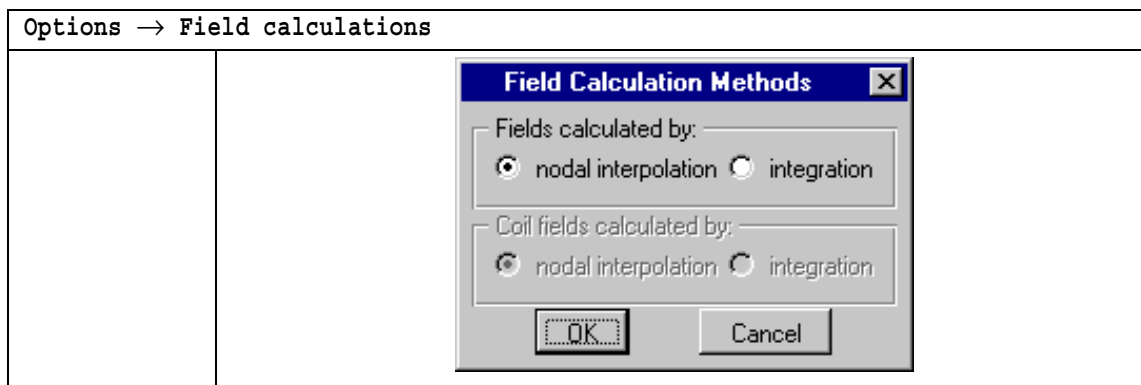
<p>Select Units icon</p> 	<p>Click on SI Units followed by OK. Note that the units in the graphics window are updated.</p>	
-------------------------------------------------------------------------------------------------------------------	----------------------------------------------------------------------------------------------------------------	--------------------------------------------------------------------------------------

It is necessary to re-display the geometry in the new units. The material colour display is also restored.



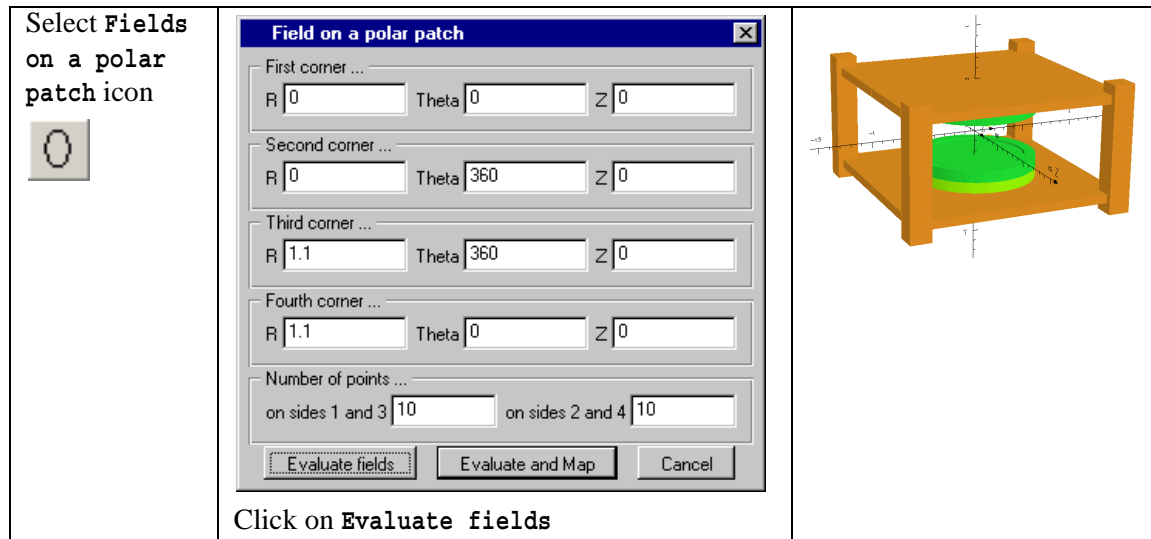
Field Values on a Plane

Before defining the surface over which the integration for the force will be performed, the field calculation method is re-set to nodally averaged fields.

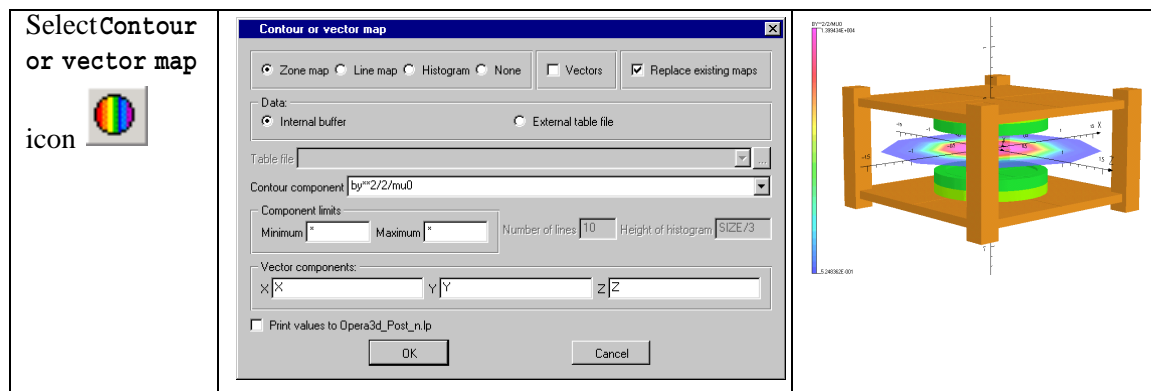


It is important that the integration captures all the flux that is passing between the poles across the mid-plane. A circular plane that extends nearly to the legs of the MRI achieves this. The user defines the circular plane in local (r, θ, z) coordinates.

The local coordinate system is still set with its local Z-axis along the global Y-axis, which is correct for the orientation of this model. Define the **POLAR** patch as follows:

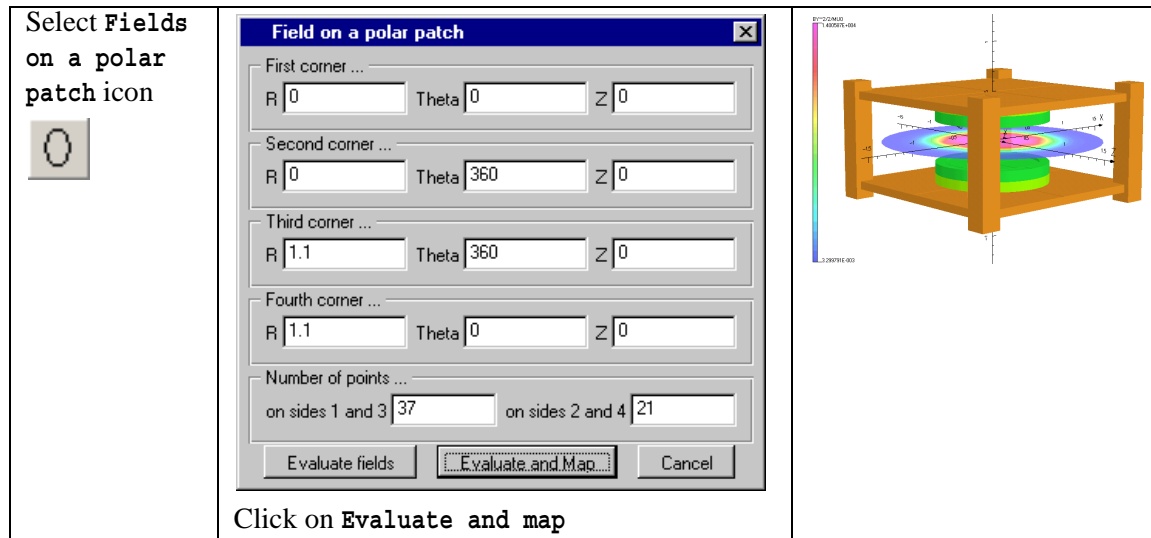


The force density, $(B_y)^2/(2\mu_0)$, can now be plotted as a contour (zone) map. This expression is written as `by**2/2/mu0`. The integral of the force density will also be returned.



At the bottom of the graphics screen, a message shows the maximum and minimum values of the component on the map (force density) and its integral (about 14290N).

The number of evaluation points on the patch is very small (10 x 10). A more accurate result will be obtained by increasing this.



The force integral increases to about 15310 N. To be confident of the result, the user should continue to increase the number of evaluation points on the patch until the answers converge to an acceptable accuracy.

File → Exit

ends the post processor session.

Chapter 5

Geometric Modeller Features

Introduction

A simple model has already been built using the modeller, and various features have been used in doing so. In this chapter, a fuller description of many of the features within the modeller is introduced. Simple examples are used to describe the various options available.

Cells and Bodies

A model is formed by creating a set of bodies and then assembling these bodies using union, intersection and subtraction operations. Initially each body consists of a single cell. Material labels and properties are attached to the cells.


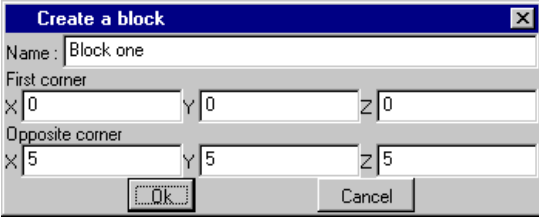
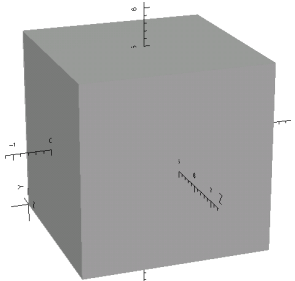


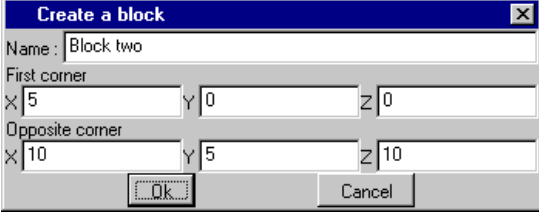
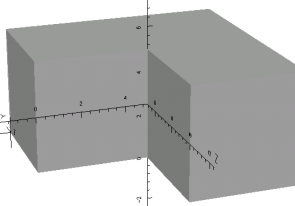

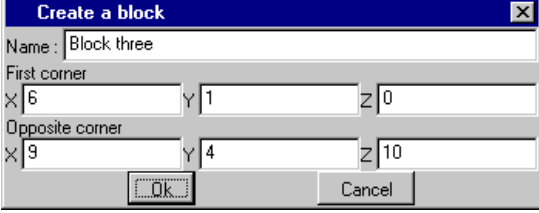
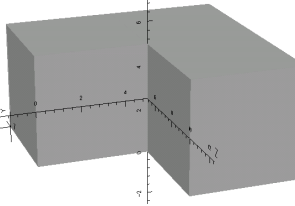
When two bodies are joined together, this creates a new body which will contain one cell if two touching bodies are unioned “with regularisation”, or two cells if the union is performed “without regularisation”. The data level of the cells determines which cell properties are applied to the resultant body parts.

For a complete description of the Modeller’s operation, see the OPERA-3d Reference Manual.

Picking and Hiding Entities



Picking entities allows the user to select entities in order to perform operations on single entities or groups, for example Boolean operations or assigning material properties. Entities can also be hidden from view to allow views of other parts of a model to be seen.

For these examples a set of 3 bodies is used to demonstrate the principles of picking and hiding. These are created in the following way:


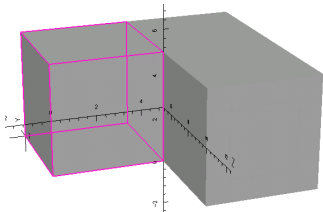
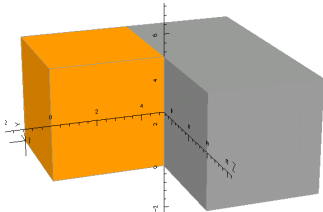
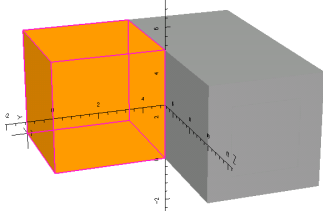
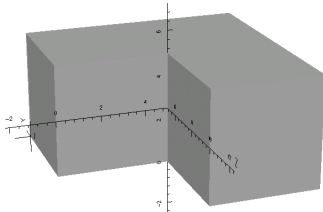
<p>Create a Block</p> 		<p>This creates the first entity.</p> 
<p>Create a second Block</p>  <p>then initialise the view</p> 		<p>This creates the second entity.</p> 
<p>Create a third Block</p> 		<p>This creates the third entity,</p> 

Note that the third block sits within block two, and as such it is not possible to see it at this stage.

Selecting Bodies


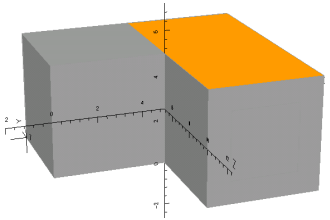

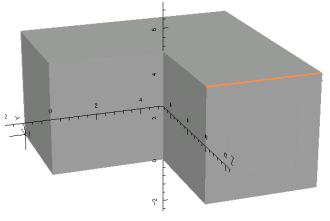

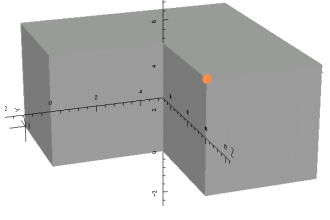
Using the example model that has just been created the bodies can be picked by first selecting the **Pick entity** button  and then selecting the **Pick bodies** button .

Now when the mouse is moved over the model, the separate bodies are highlighted by a coloured edging. Once a body is highlighted it can be selected by double clicking the left mouse button.

Pick a body 	Moving the mouse over Block one highlights the body with a pink edge.	Body edges are highlighted in pink 
	Double clicking the left mouse button selects Block one .	Selected bodies are coloured orange. 
	Moving the mouse over Block one highlights the body again.	
	Double clicking the left mouse button de-selects Block one .	

Selecting other Entities

In exactly the same way the cells, edges and vertices of the bodies can also be selected.


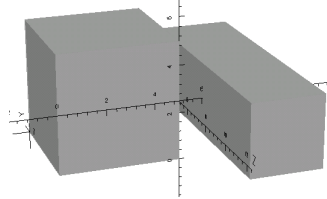
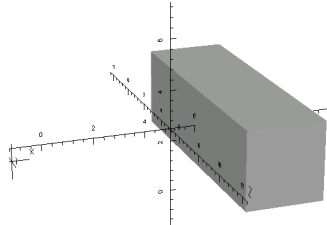

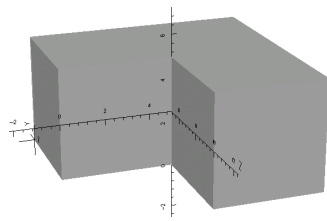

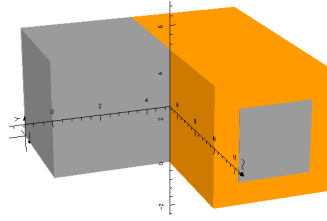

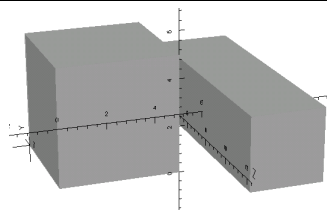
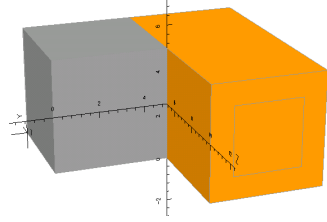
<p>Pick a face</p> 	<p>A face can be selected or de-selected with the Pick faces button and then double clicking the left mouse button on the desired face. Faces will be highlighted in the same way that cells and bodies are.</p>	
<p>Pick an edge</p> 	<p>An edge can be selected or de-selected with the Pick edges button and double clicking on the edge required.</p>	
<p>Pick a vertex</p> 	<p>A vertex can be selected in exactly the same way.</p>	

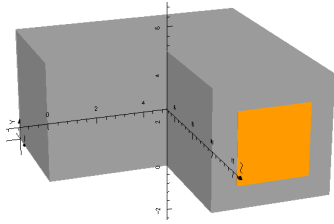
Entities can be selected singly, as has been shown in the above example, or by repeating the selection process more than one entity of the same type can be selected. Using multiple selections allows the mesh properties or material labels to be assigned to more than one entity at a time. Also when performing Boolean operations primitives can be joined to form more complex bodies.

Hiding Entities


One of the three blocks that has been created in the above example is not easily accessible for picking or viewing, so there are two ways in which some entities can be hidden from view to facilitate this. The first method will hide an entity until the **Unhide entities** button is pressed. In this way several entities can be hidden from view and then made visible again with a single button push. The second method is to temporarily hide a picked entity until the next action is completed, whereupon the hidden entity will become visible again. This is most useful when having to select bodies in a particular order to get the correct result from a

Boolean operation and will be explained further in the section on Boolean operations.

<p>Hide several bodies</p> 	<p>Double clicking on the bodies within the model as they are highlighted will cause the picked body to become hidden from view. Picking Block two will now hide it from view making Block three visible.</p>	
	<p>Picking Block one will also hide it from view leaving just Block three visible.</p>	
<p>Un-hide hidden entities</p> 	<p>By clicking on the Unhide entities button all of the hidden entities are made visible again.</p>	
<p>Hiding picked entities</p> 	<p>The other method of hiding entities is to hide an entity that has already been selected using the Pick entity button and the relevant entity button. If Block two is selected as before the front face of block three can be partly seen as it lies on the same surface as the front face of Block two.</p>	
<p>Hide picked entities</p> 	<p>By now pressing the Hide picked entities button Block two will be temporarily hidden.</p>	
	<p>Now Block three can be selected. In doing this Block two now become visible again.</p>	

	<p>Block two can now be de-selected by simply double clicking on it again when it is highlighted. This leaves Block three as the only picked entity and now its front face can be seen on the front face of Block two.</p>	
--	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-------------------------------------------------------------------------------------

Reset Picked Entities

Once a set of entities has been picked and are highlighted in orange they can be reset by simply pressing the **Reset picked entities** button .

Before moving to the next section, clear all the data using:

FILE ↓
 Close


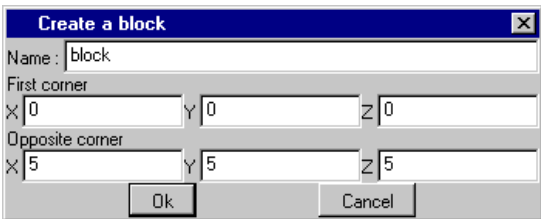
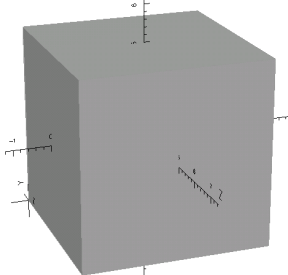

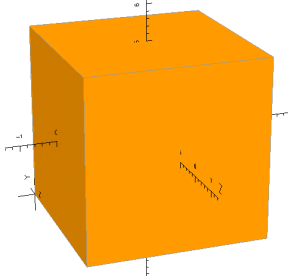
Display Options

All objects created in the Modeller are composed of different types of entity including bodies, cells, faces, edges and vertices. The relationship between these entities is explored in this section. Entity selection and display along with the display of associated properties is also shown.

In this example a simple cube is created. The entities that make up the cube and the properties associated with the entities are then investigated.

Bodies


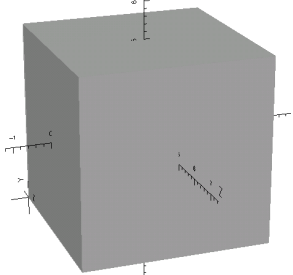

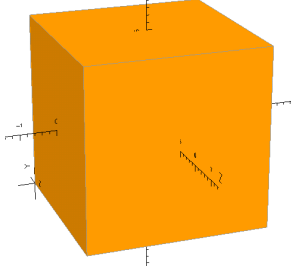
The cube is created as a body. Bodies are picked to allow **Operations** such as Boolean combination, replication or transformation.

<p>Create a block</p> 		
<p>Pick the block as a body</p> 	<p>Move the mouse on to the block. Double click to select the object. Once selected the object will turn orange.</p>	

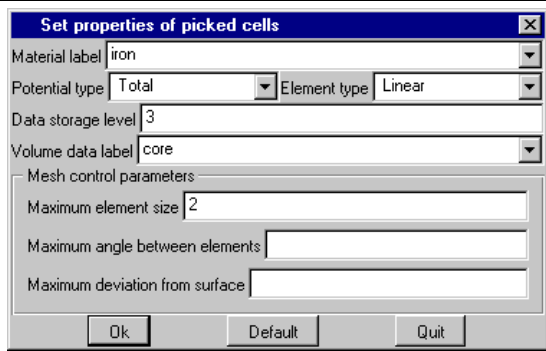
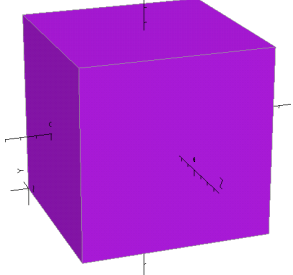
Note that the body entity is a hierarchical collection of cells, faces, edges and vertices. This means that once a body has been picked it can be moved or copied to create other bodies. (See [“Transforming Objects” on page -1](#)). This will not be done in this section. A body has no properties associated with it. Only cells, faces, edges and vertices can have properties via the labels assigned to them. Labels are assigned to entities, and property values assigned to labels.

Cells

A body is generally made from one or more cells - although this is not mandatory. Any collection of geometric entities may constitute a body.

Reset the picked entity 	Double click the picked cube to deselect it or use the Reset picked entities icon.	
Pick cells 	Double click on the cube.	

Set the properties of the selected cells.

Properties → Cell properties		
		 Note: This colour may not be the same as on your screen.


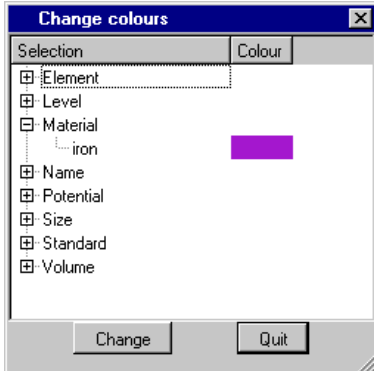
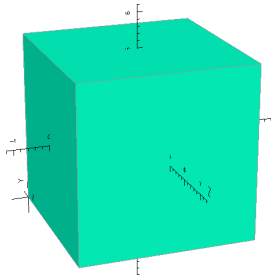
The **Material** and **Volume** label names are supplied by the user. The labels can be used throughout the model. The **Material** label is used to apply material properties etc. The **Volume** label is used to apply velocities, current densities, packing factors and charge densities if necessary. The **Material** and **Volume** properties themselves are set in the top **Properties** menu of the modeller. Once assigned a

material name, other cells with the same material properties are the same colour. This can be changed (see below).

The **Data storage level** is a number supplied by the user to indicate which properties take priority when cells are joined or merged. The higher the number the higher the priority.


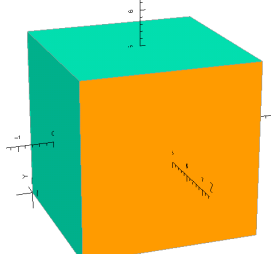
The **Mesh control parameters** are used to control the mesh around a cell and are discussed elsewhere.

The colour used for a particular material can also be changed.

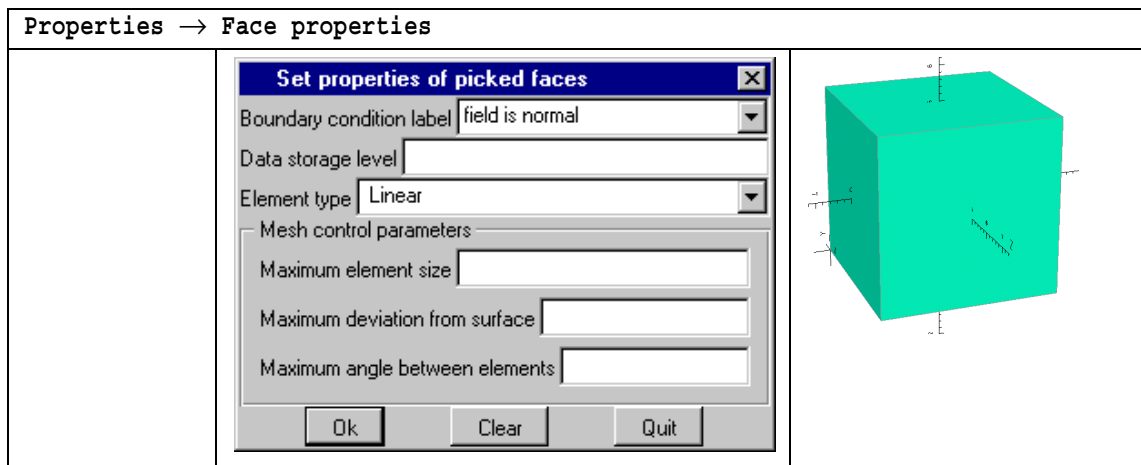
<p>Change the colours</p> 	 <p>Click on iron and then change. A colour selector will appear where you can choose a new colour. Enter Quit to leave the menu.</p>	<p>Example of new colour.</p> 
-------------------------------------------------------------------------------------------------------------	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--------------------------------------------------------------------------------------------------------------------

Faces

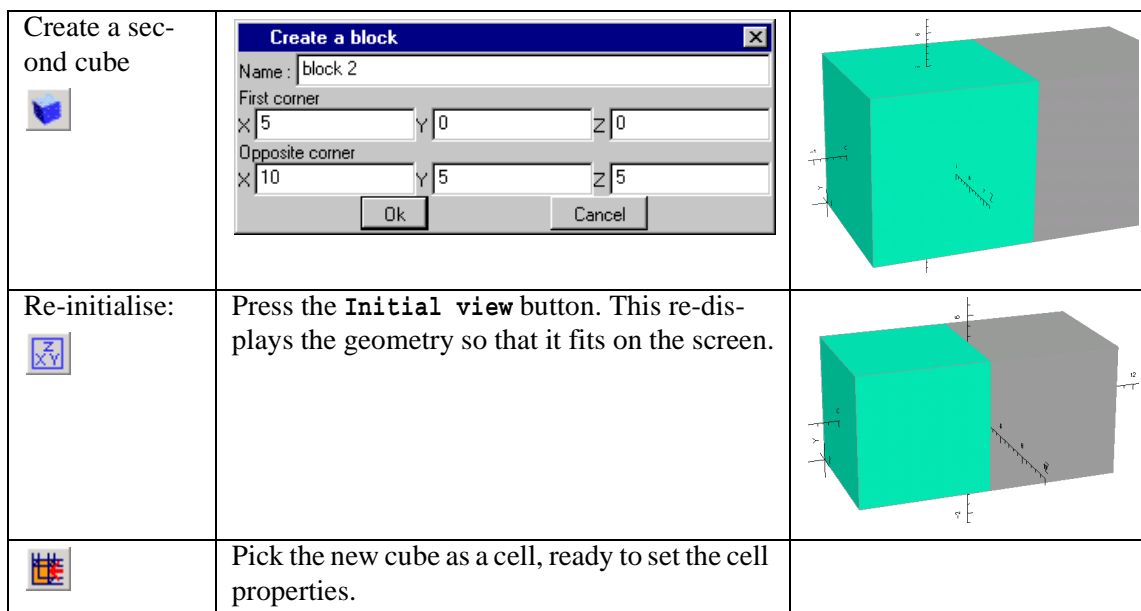
Faces are two dimensional geometric entities.

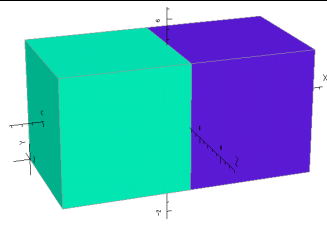

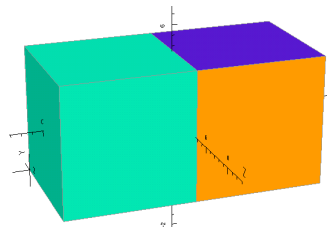

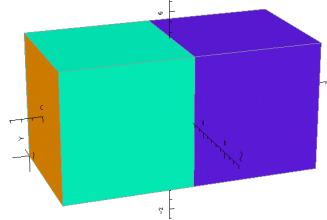
<p>Pick faces</p> 	<p>Select the face in order to change its properties (applying boundary conditions for example).</p>	
-------------------------------------------------------------------------------------------------------	------------------------------------------------------------------------------------------------------	---------------------------------------------------------------------------------------

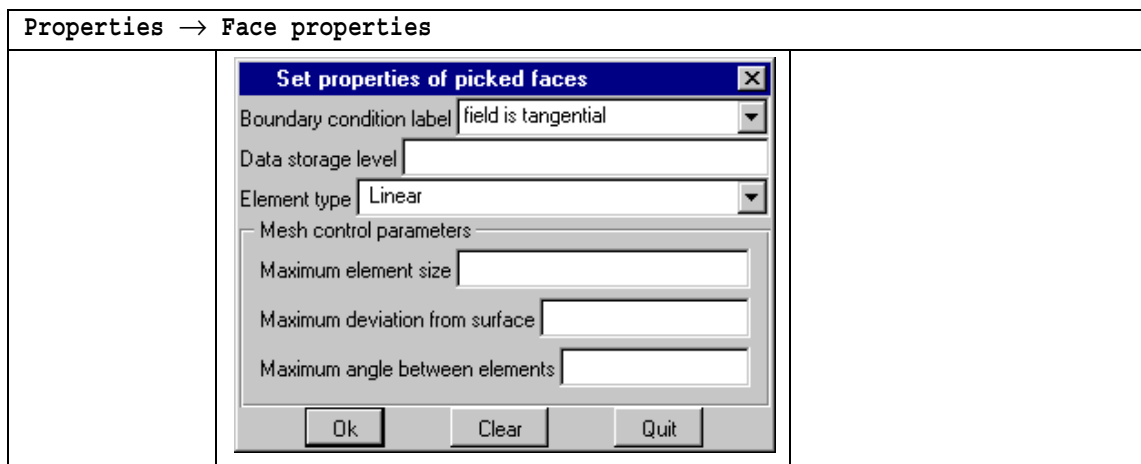
Set the face properties:



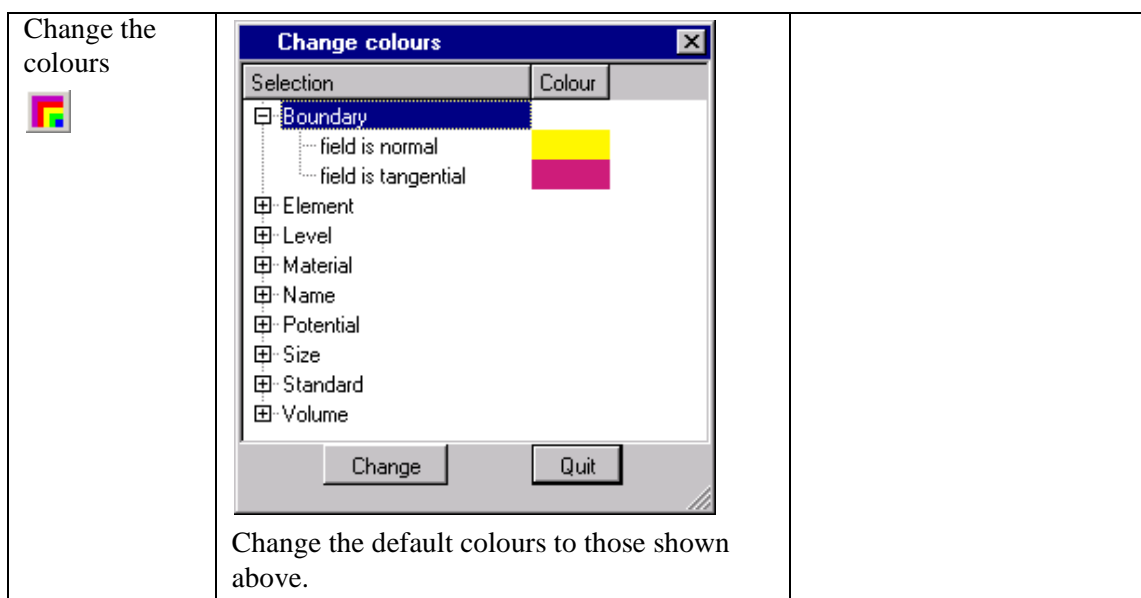
The **Boundary condition label** is used to assign boundary conditions to the model. The label can be used anywhere in the model and as many times as required. The boundary conditions themselves are set in the **Model → Set boundary conditions** menu of the modeller.


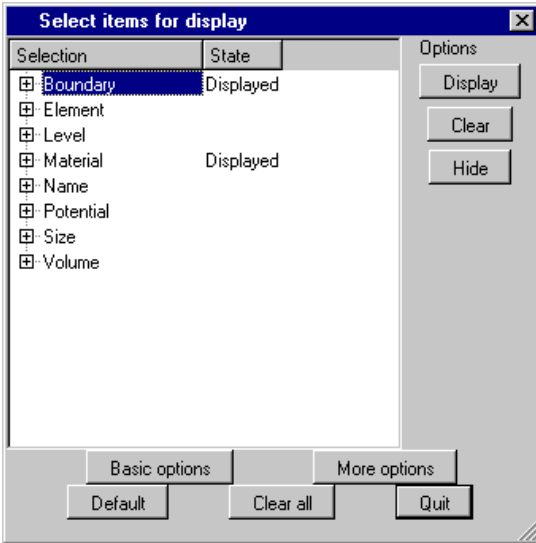
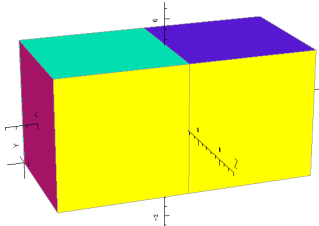


Properties → Cell properties		
	Set its cell properties to Material steel . Leave the other cell properties as default.	
Pick faces: 	Pick the face on the new block as shown.	
Properties → Face properties		
	Use the arrow at the end of the Boundary condition label box to obtain a list of labels already used.	
Pick faces: 	Select the face at the end of the first block, and set its face properties.	



The boundary colours can be changed, as follows:


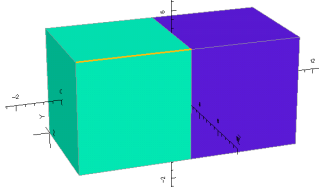



<p>Change the selection:</p> 	 <p>Make sure field is normal and field is tangential are set to be displayed as shown above.</p>	
----------------------------------------------------------------------------------------------------------------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-------------------------------------------------------------------------------------

Use **Clear** in the dialog box to remove the boundary conditions display.

Edges

Each face is made from a number of edges.

<p>Pick edges:</p> 	<p>Select any edge on either block, ready to set the edge properties.</p>	
<p>Properties → Edge properties</p>		
	<p>The edge properties are used as an extra mesh control. Where 2 edges are coincident, the Data storage level affects which mesh control size takes priority. The higher the number, the higher the priority.</p> 	

Vertex properties may also be used for **Mesh control**. Test this by picking any vertex in the model and opening up the **Vertex properties** dialog box.

Before moving to the next section, clear all the data using:


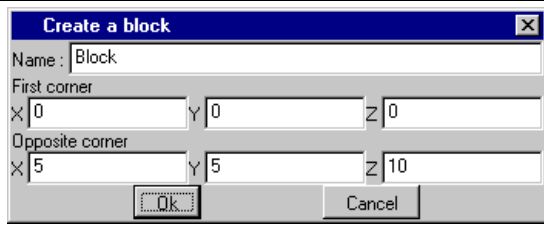
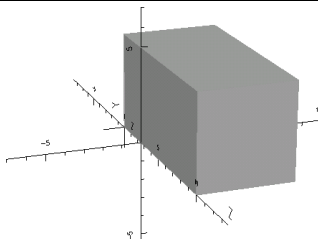

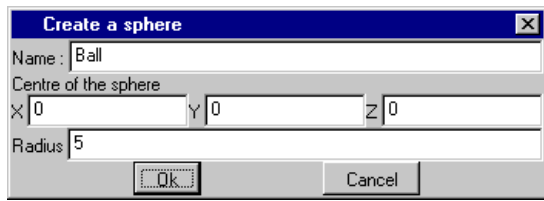
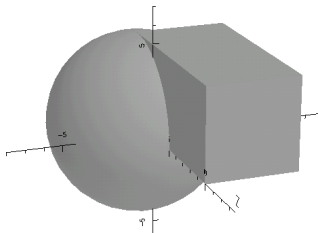

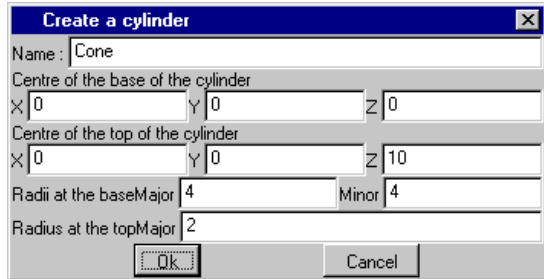
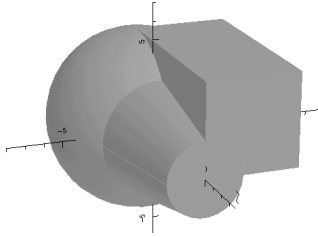
FILE ↓
 Close

Boolean Operations

Boolean operations allow the user to combine simple primitives to build a solid model geometry. Primitives can be combined in such a way as the resultant body can be a union of the starting bodies or the bodies can be used to subtract or trim regions from the initial geometry.


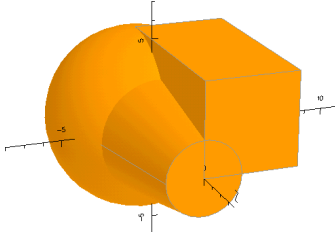
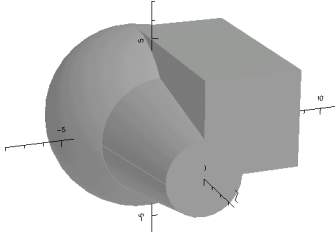
The five basic operations are Union, Intersection, Subtraction, Trim Overlap and Cutaway Overlap. The Union, Subtraction, Trim Overlap and Cutaway Overlap operations can then be applied with or without Regularisation, which is the process to remove overlapping or internal cells, faces, edges and vertices from the resultant body.


In order to explain these operations a set of examples follow that go through each of the seven options available to the user. For these examples three bodies are used to demonstrate the principles which are created in the following way:

<p>Create a Block</p> 	 <p>Create a block</p> <p>Name : Block</p> <p>First corner</p> <p>X 0 Y 0 Z 0</p> <p>Opposite corner</p> <p>X 5 Y 5 Z 10</p> <p>Ok Cancel</p>	
<p>Create a Sphere</p> 	 <p>Create a sphere</p> <p>Name : Ball</p> <p>Centre of the sphere</p> <p>X 0 Y 0 Z 0</p> <p>Radius 5</p> <p>Ok Cancel</p>	
<p>Create a Cone</p> 	 <p>Create a cylinder</p> <p>Name : Cone</p> <p>Centre of the base of the cylinder</p> <p>X 0 Y 0 Z 0</p> <p>Centre of the top of the cylinder</p> <p>X 0 Y 0 Z 10</p> <p>Radii at the baseMajor 4 Minor 4</p> <p>Radius at the topMajor 2</p> <p>Ok Cancel</p>	

Union With Regularisation


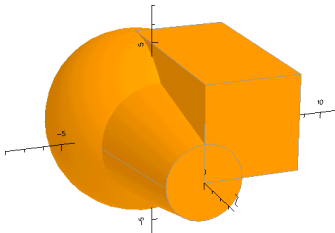
The first operation to look at is the **Union With Regularisation**. This allows several bodies to be combined to form a single body which consists of a single cell.

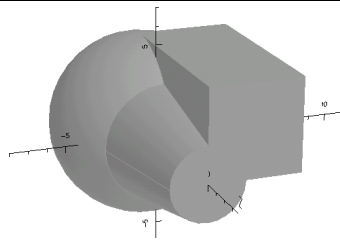


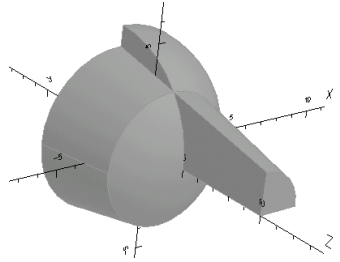
Pick bodies 	Using the mouse select each of the three bodies by double clicking the left mouse button as each body is highlighted by a pink edging.	
Operations → Combine bodies → Union with regularisation		
	The three bodies are now combined to form a single body which consists of a single cell. The status bar at the bottom of the Modeller window now shows that the model consists of 1 body and 1 cell.	

The **Undo** button  can be used to return the solid model back to the point before the **Union** was applied, or the primitives can be re-created to allow the next example to be completed.

Union Without Regularisation


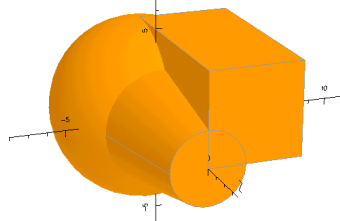
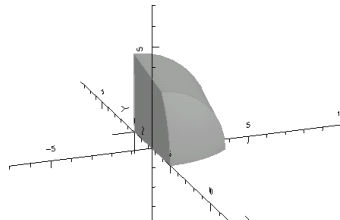
The next operation is the **Union Without Regularisation**. This command allows two or more bodies to be joined together such that the original cells are kept and any intersections of the original cells will form new cells. The new cells created can then have separate properties assigned to them to allow control of mesh density.

Pick bodies 	Using the mouse select each of the three bodies by double clicking the left mouse button as each body is highlighted.	
-----------------------------------------------------------------------------------------------------------	-----------------------------------------------------------------------------------------------------------------------	---------------------------------------------------------------------------------------

Operations → Combine bodies → Union without regularisation		
	<p>The three bodies are now combined to form a single body. However as the without regularisation option was chosen the overlap of the original cells has created a set of new cells within the body. This can be seen by looking at the status bar at the bottom of the Modeller window which will now be showing 1 body and 7 cells.</p>	
<p>Hide cells</p> 	<p>The new cells can be seen by hiding the original cells that formed the Block, Sphere and Cone. Double clicking the left mouse button as each of these cells is highlighted will hide these cells until the Unhide entities button  is pushed. In this way it is possible to view each of the new cells that was formed from an intersection of the original cells.</p>	


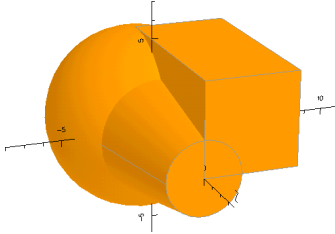
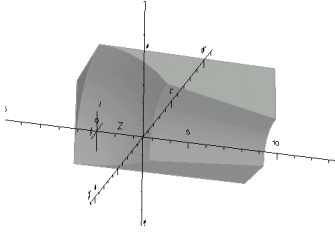

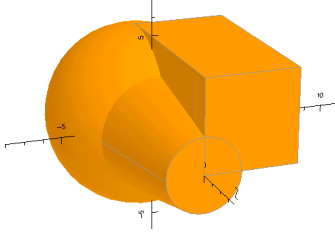
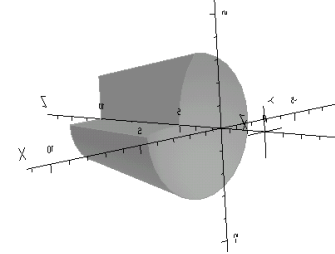
Intersection With Regularisation

The next type of operation that can be completed is an **Intersection**. This results in a new body being created that is formed from the intersection of the starting bodies.

<p>Pick bodies</p> 	<p>Using the mouse select each of the three bodies by double clicking the left mouse button as each body is highlighted.</p>	
Operations → Combine bodies → Intersection with regularisation		
	<p>The resultant single body is formed from the intersection of the original three bodies. The parts of the original bodies that were not intersecting are removed. Again the status bar shows that there is now only a single body which comprises one cell.</p>	


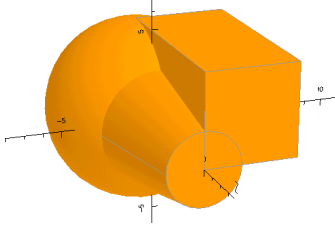
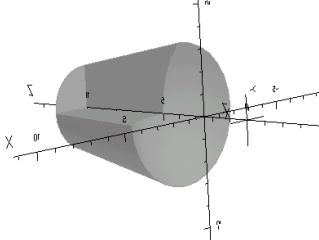
Subtraction With Regularisation

The **Subtraction** operation gives different results depending on the order in which the bodies are selected. The first body picked will have the intersections of the other selected bodies removed from it and the other selected bodies will be removed.

Pick bodies 	Using the mouse select each of the three bodies by double clicking the left mouse button as each body is highlighted. Start by picking the Block first and then the Cone and Sphere.	
Operations → Combine bodies → Subtraction with regularisation		
	The body that results from this operation is formed from subtracting the Cone and Sphere from the Block.	
Pick bodies 	Using the mouse select each of the three bodies by double clicking the left mouse button as each body is highlighted. Start by picking the Cone first and then the Block and Sphere.	
Operations → Combine bodies → Subtraction with regularisation		
	By picking the bodies in a different order a very different resultant body will be formed. The subtraction operation subtracts the second and third body from the first.	

Subtraction Without Regularisation

The **Subtraction** operation can also be applied without regularisation. The resultant body will then consist of a subtraction of the second and subsequent bodies from the first one selected plus any faces that lie on the surface of the removed parts. This allows the user to leave behind faces that can then have mesh parameters set on them.


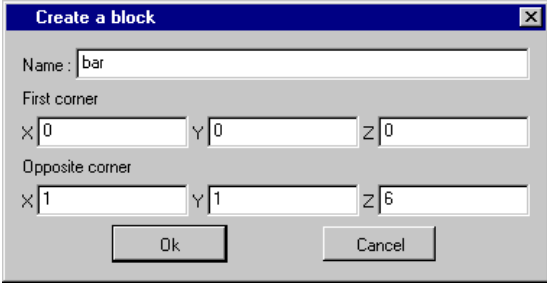
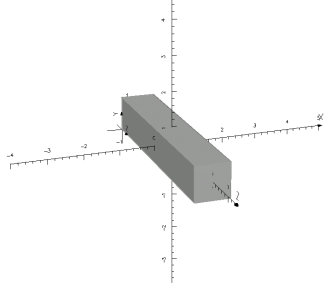


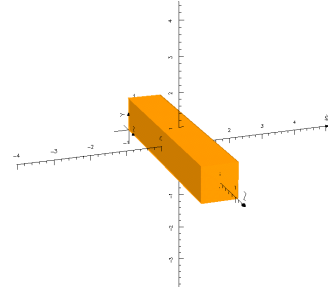
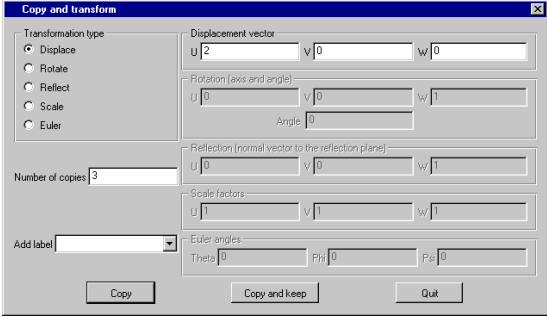
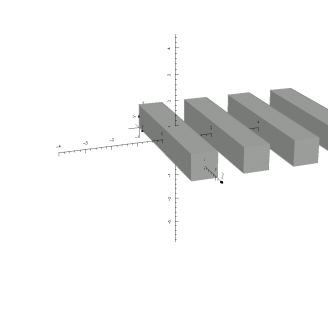

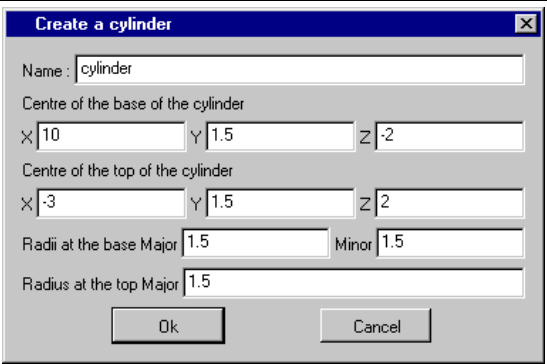
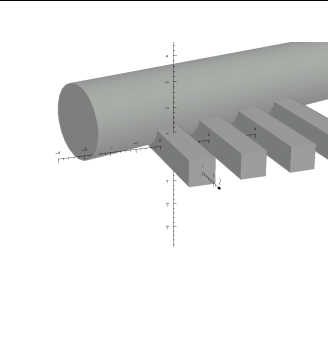
Pick bodies 	Using the mouse select each of the three bodies by double clicking the left mouse button as each body is highlighted. Start by picking the Cone first and then the Block and Sphere.	
Operations → Combine bodies → Subtraction without regularisation		
	The subtraction operation subtracts the second and third body from the first but as the bodies are not regularised the face that lay on the surface of the block on the end of the cone has been left. This face can now be used to apply local mesh refinement etc. as it can have properties assigned to it.	


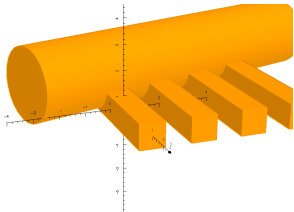
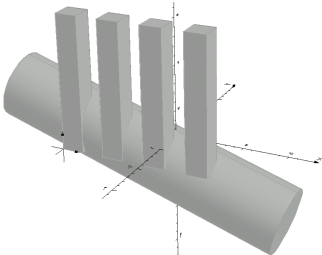

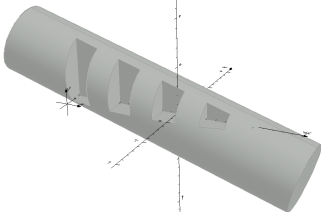
Trim Overlap With Regularisation

Another type of operation is **Trim Overlap** which allows the user to remove parts of bodies that overlap and still leave the original bodies. In effect the command issues a **Subtraction** but taking a copy of the original bodies first and replacing these once the **Subtraction** is complete.

The **Trim Overlap** operation is very useful when adding volumes around regions of interest so that the mesh can be controlled more easily. A new body can be created that overlaps an area that requires a fine mesh. If this is picked first, followed by the other bodies that overlap, then when the **Trim Overlap** command is issued the new volume will fill the gaps around the area of interest. This new volume can

then have a mesh parameter assigned such that a finer mesh is produced only in this area.

<p>Create a Block</p> 		
<p>Pick bodies</p>  	<p>Select the block by double clicking with the left mouse button.</p>	
<p>Operations → Copy</p>		
		
<p>Create a Cylinder</p> 		

Pick bodies 	Using the mouse select the cylinder first, then the four bars.	
Operations → Combine bodies → Trim Overlap with regularisation		
	The resultant geometry still contains 5 bodies, but the overlap between the cylinder and the 4 bars has now been removed.	
Hide bodies 	By now hiding the 4 bars one by one the result of the operation can be seen more clearly. The 4 bars have been cut out of the cylinder.	


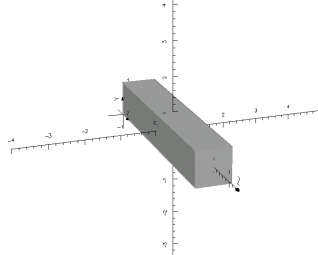

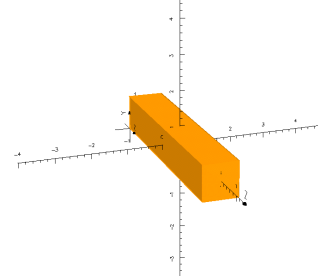
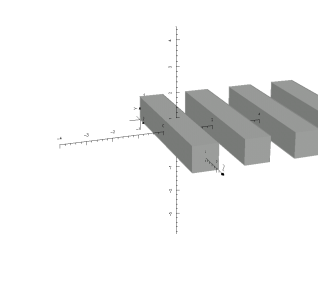

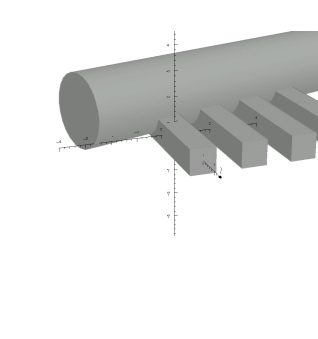
Trim Overlap Without Regularisation


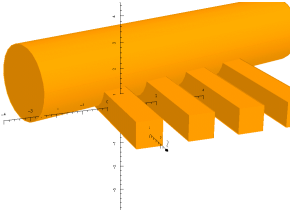
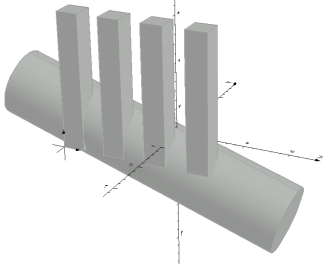

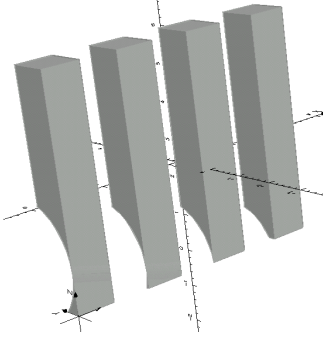
The **Trim Overlap** operation can also be applied without regularisation. This produces a similar result to that achieved in the **Subtraction** operation in that the intersecting faces are left on the surface of bodies after the operation is completed. Repeating the above example produces exactly the same result as the operation with regularisation, because no faces are shared between the cylinder and the 4 bars. The operation can be tried out quickly by pressing the **Undo** button twice, which reverts to the status before the last operation, and then applying the **Trim Overlap without regularisation**.

Cutaway Overlap With Regularisation

The last type of operation is **Cutaway Overlap** which is very similar to the previously discussed **Trim Overlap** operation. The difference is that the cylinder of the above example is used as a cutting tool to cut the overlapping parts from the 4 rectangular bars. The operation is useful when several entities shall be trimmed in one

step. Again the cylinder has to be picked first, followed by the other bodies that overlap, then the **Cutaway Overlap** command can be issued.

<p>Create a Block</p> 	<p>Create a block</p> <p>Name : bar</p> <p>First corner</p> <p>X 0 Y 0 Z 0</p> <p>Opposite corner</p> <p>X 1 Y 1 Z 6</p> <p>Ok Cancel</p>	
<p>Pick bodies</p> 	<p>Select the block by double clicking with the left mouse button.</p>	
<p>Operations → Copy</p>		
	<p>Copy and transform</p> <p>Transformation type</p> <p><input checked="" type="radio"/> Displace</p> <p><input type="radio"/> Rotate</p> <p><input type="radio"/> Reflect</p> <p><input type="radio"/> Scale</p> <p><input type="radio"/> Euler</p> <p>Displacement vector</p> <p>U 2 V 0 W 0</p> <p>Rotation (axis and angle)</p> <p>U 0 V 0 W 1</p> <p>Angle 0</p> <p>Reflection (normal vector to the reflection plane)</p> <p>U 0 V 0 W 1</p> <p>Scale factors</p> <p>U 1 V 1 W 1</p> <p>Euler angles</p> <p>Theta 0 Phi 0 Psi 0</p> <p>Number of copies 3</p> <p>Add label</p> <p>Copy Copy and keep Quit</p>	
<p>Create a Cylinder</p> 	<p>Create a cylinder</p> <p>Name : cylinder</p> <p>Centre of the base of the cylinder</p> <p>X 10 Y 1.5 Z -2</p> <p>Centre of the top of the cylinder</p> <p>X -3 Y 1.5 Z 2</p> <p>Radii at the base Major 1.5 Minor 1.5</p> <p>Radius at the top Major 1.5</p> <p>Ok Cancel</p>	

<div>Pick bodies</div> <div></div>	<div>Using the mouse select the cylinder first, then the four bars.</div>	
<div>Operations → Combine bodies → Cutaway Overlap with regularisation</div>		
	<div>The resultant geometry still contains 5 bodies but the overlapping parts between the cylinder and the 4 bars have been cut away from the 4 bars.</div>	
<div>Hide bodies</div> <div></div>	<div>By now hiding the cylinder, the result of the operation can be seen more clearly. The cylinder has been left untouched, whereas the 4 bars have been trimmed.</div>	

Cutaway Overlap Without Regularisation

The **Cutaway Overlap** operation can also be applied without regularisation. Similar to the previous example this operation produces the same result to that achieved in the operation with regularisation, as no faces are shared between the cylinder and the 4 bars. The operation can be tried out quickly by pressing the **Undo** button twice, which reverts to the status before the last operation, and then applying the **Cutaway Overlap without regularisation**.

Before moving to the next section, clear all the data using:

FILE ↓
Close


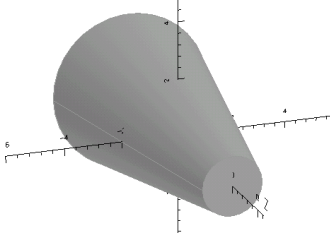

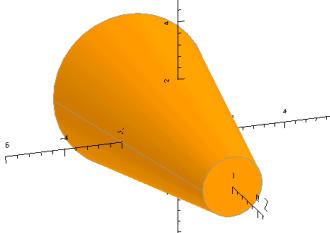
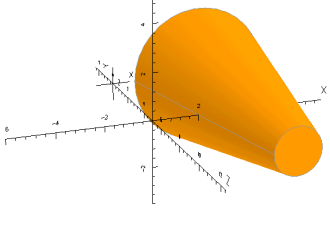
Transforming Objects

Objects can be moved or copied in a number of different ways. The **Copy** operation allows multiple instances of the object to be created.

In this example, a simple cone is created. The **Transformation** and **Copy** operations are then demonstrated.

Transformations

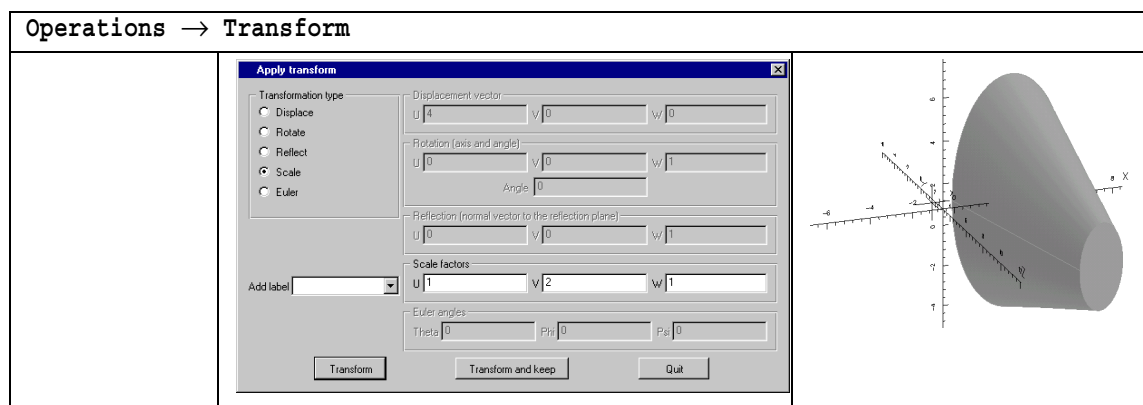
A **body** can be created in any convenient coordinate system and then moved to its correct position later using **Transformation**. **Transformation** can also be used to change the shape of a **body**.

<p>Cylinder/Cone</p> 	<p>Create a cylinder</p> <p>Name : cone</p> <p>Centre of the base of the cylinder</p> <p>X 0 Y 0 Z 0</p> <p>Centre of the top of the cylinder</p> <p>X 0 Y 0 Z 10</p> <p>Radii at the base Major 3 Minor 3</p> <p>Radius at the top Major 1</p> <p>Ok Cancel</p>	
<p>Pick bodies</p> 	<p>Move the mouse on to the cone. Double click to select the object. Once selected the object will turn orange.</p>	
<p>Operations → Transform</p>		
	<p>Apply transform</p> <p>Transformation type</p> <p><input checked="" type="radio"/> Displace</p> <p><input type="radio"/> Rotate</p> <p><input type="radio"/> Reflect</p> <p><input type="radio"/> Scale</p> <p><input type="radio"/> Euler</p> <p>Displacement vector</p> <p>U 4 V 0 W 0</p> <p>Rotation (axis and angle)</p> <p>U 0 V 0 W 1</p> <p>Angle 0</p> <p>Reflection (normal vector to the reflection plane)</p> <p>U 0 V 0 W 1</p> <p>Scale factors</p> <p>U 1 V 1 W 1</p> <p>Euler angles</p> <p>Theta 0 Phi 0 Psi 0</p> <p>Add label</p> <p>Transform Transform and keep Quit</p>	


Use **Transform** and **keep**, to keep the object selected.

Changing the shape of a body

Under the **Transform** options, the shape of the body may also be altered using scaling.

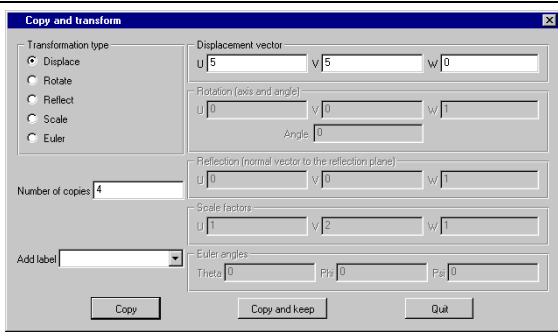
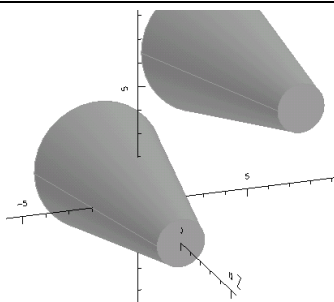



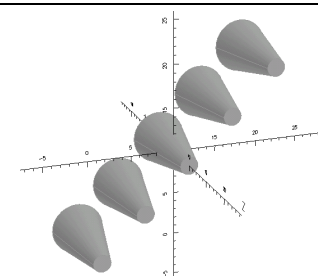
Use **Transform** this time, to deselect the object after making the transformation.

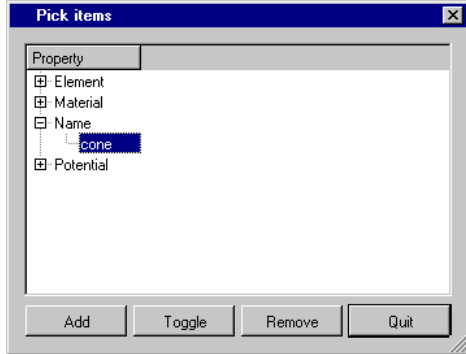
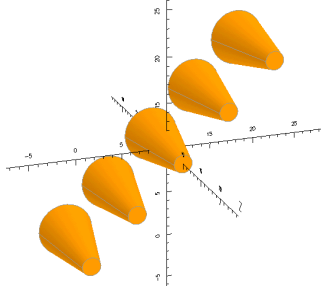
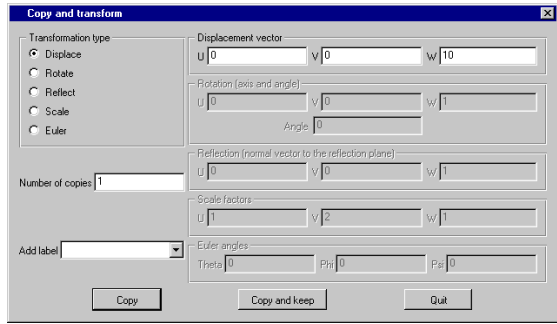
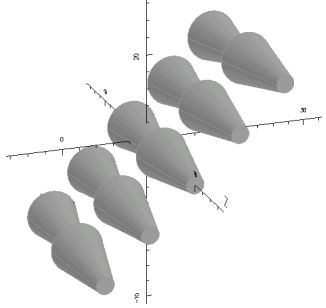

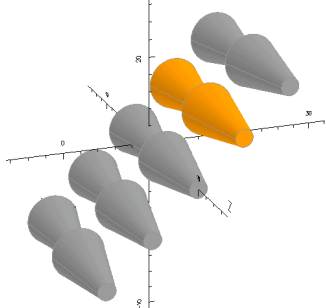
Select the **Undo** icon  once (or use **Edit** → **Undo**) to return the picked (orange) cone to its original position. Note that even though two separate transformations (**Displace** and **Scale**) have been used, this counts as one operation when **Undo** is used.

Making Replications of a Body

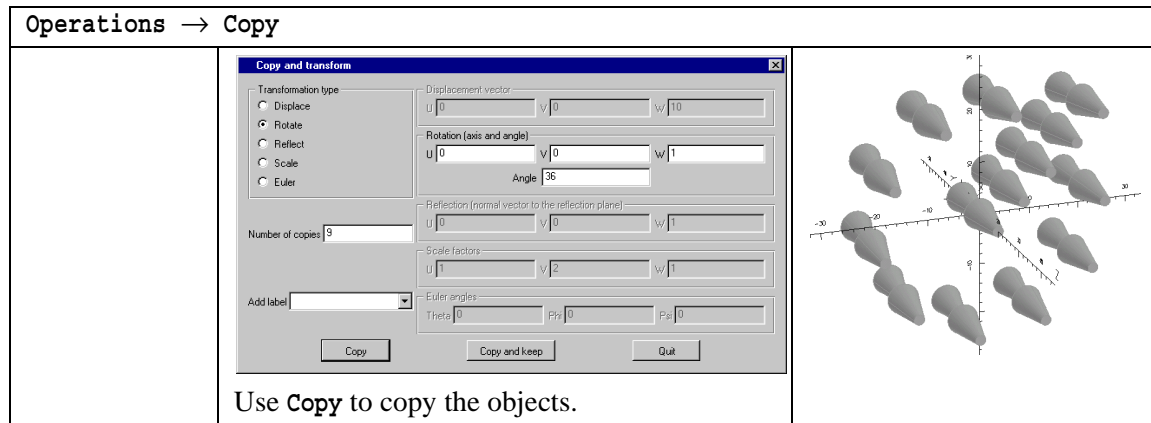
Many complex geometric models consist of repeated copies of a simple subset of objects. A single copy of the subset can be defined and multiple replications made.

Operations → Copy		
	 <p>Use Copy to replicate the object and deselect the original cone.</p>	 <p>Note that the Number of copies is set to 4 i.e. 4 new copies are made.</p>

View → Views → Initial view		
<p>or use the icon:</p> 		

Picking → Pick entities by property		
	<p>The model now contains 5 bodies, but all are called cone.</p>  <p>Click on Add to pick all the bodies.</p>	
Operations → Copy		
	 <p>Use Copy to make one copy of the objects.</p>	
<p>Pick bodies</p> 	<p>Pick 2 cones to produce the selections shown.</p>	

The two selected cones are now copied also.



Note that it is also possible to copy **cells** and **faces**. For example, if 2 overlapping sphere **bodies** are **Combined** with a **Non-regular union**, the overlapping **cell** can be selected and copied to a new location to create a new **body**. Copying of **faces** is also possible.

Before moving to the next section, clear all the data using:

FILE ↓
Close

Sweeping Faces

As well as producing models using Boolean operations on primitives it is possible to extrude a volume by sweeping a face. This can be done in three ways; by sweeping along a straight line, either perpendicular to the face or along a vector, or by sweeping in an arc about a defined centre.

When a face is swept the sides of the extrusion can be kept parallel or they can be allowed to diverge or converge by setting a draft angle.

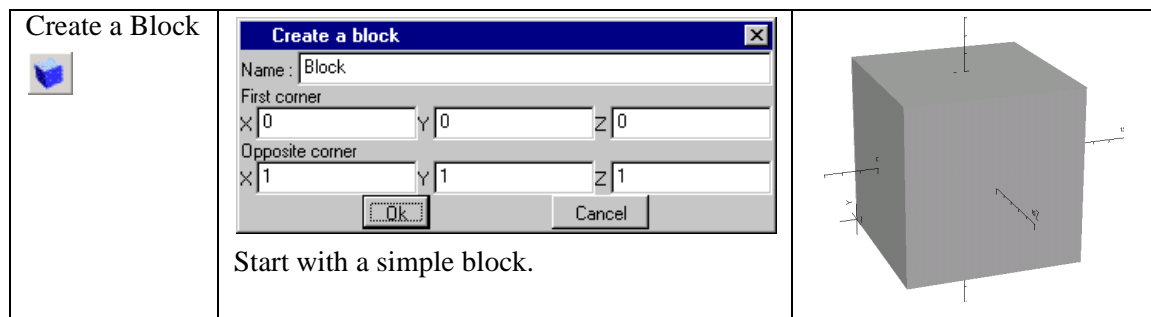
The original face can either be left in place or removed. Leaving the original face in place will mean that the extruded volume produces a new cell within the body, removing it means that the existing cell is extended.


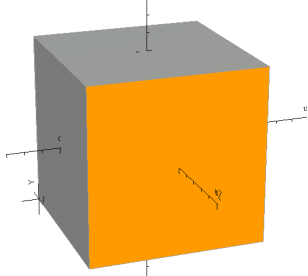
The face being swept does not have to be a simple primitive face but can be a face created from Boolean operations. However it must be a flat surface.

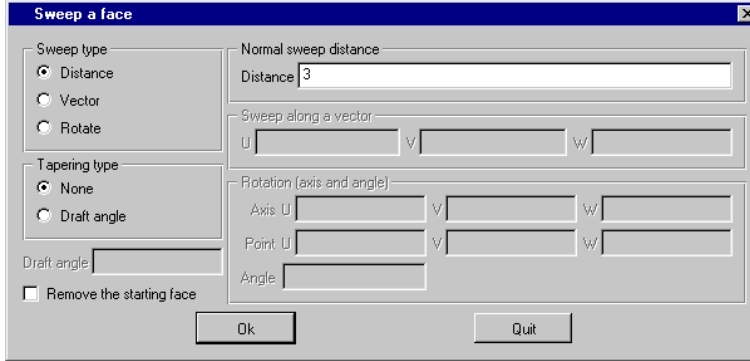
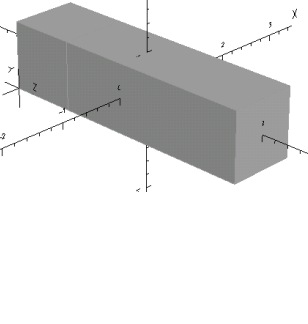
The direction of the extrusion is controlled by the face normal vector, which can be viewed by selecting **Face normal (picked faces only)** from the **View → Vectors** menu.

Sweep a Face through a Distance

Sweeping a face through a distance allows the starting face to be extruded in a linear direction that is normal to the starting face.




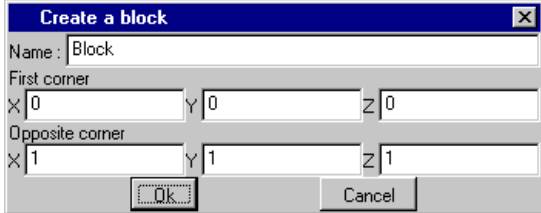
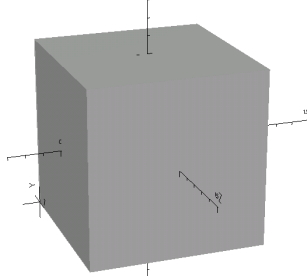
<p>Pick a Face</p> 	<p>Ensure that the Pick entity icon is pressed and then select the front face (at $z=1$) by double clicking on it when highlighted.</p>	
------------------------------------------------------------------------------------------------------	-----------------------------------------------------------------------------------------------------------------------------------------------------------	-------------------------------------------------------------------------------------


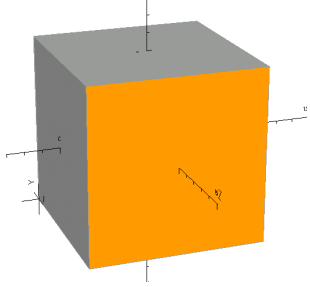
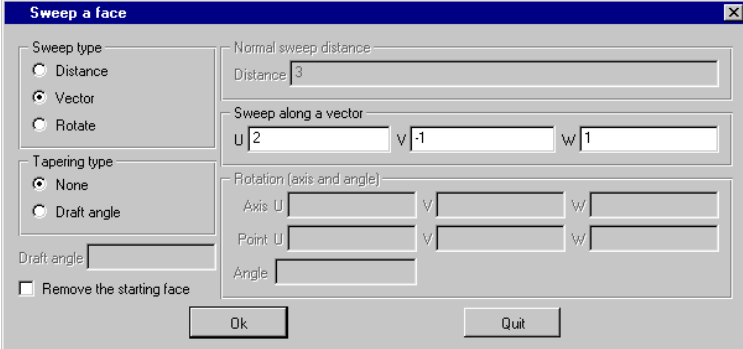
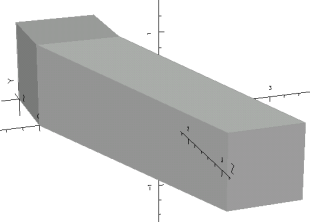
<p>Operations → Sweep face</p>		
		

By not removing the starting face, the body is now made up of two cells which could then have different material labels or meshes applied to them.

Sweep a Face along a Vector

Sweeping a face along a vector also produces a linear extrusion but this time there is more control over the extrusion direction.


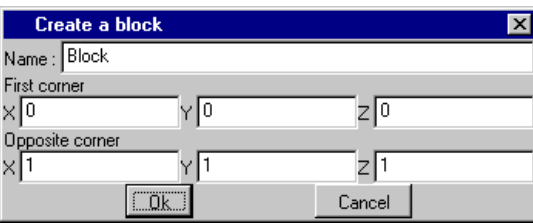
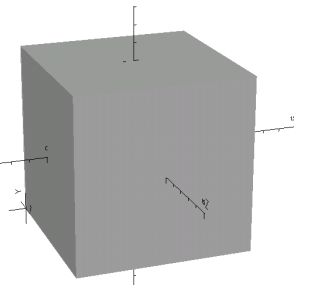
<p>Create a Block</p> 	 <p>Start with a simple block.</p>	
-----------------------------------------------------------------------------------------------------------	------------------------------------------------------------------------------------------------------------------------	---------------------------------------------------------------------------------------


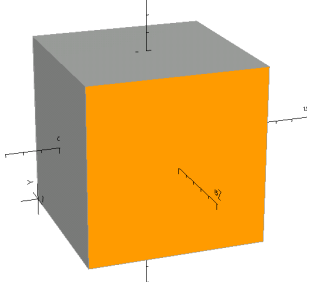
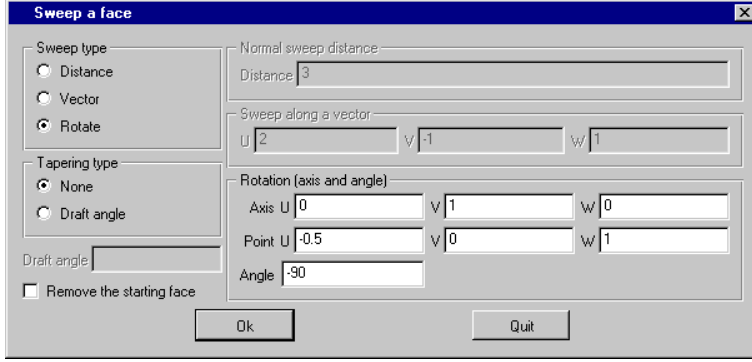
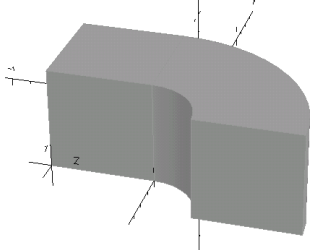
<p>Pick a Face</p> 	<p>Select the front face (at $z=1$) by double clicking on it when highlighted.</p>	
<p>Operations → Sweep face</p>		
		

The face has now been swept along the defined vector from the position of the starting face to produce the new shape. Again as the starting face has been left the body now consists of two cells.

Sweep a Face by Rotation

The **Sweep a Face by Rotation** is a much more powerful command that allows very complex shapes to be produced.

<p>Create a Block</p> 	 <p>Start with a simple block.</p>	
-----------------------------------------------------------------------------------------------------------	------------------------------------------------------------------------------------------------------------------------	---------------------------------------------------------------------------------------

<p>Pick a Face</p> 	<p>Select the front face (at $z=1$) by double clicking on it when highlighted.</p>	
<p>Operations → Sweep face</p>		
		

The face is swept through an angle of -90 degrees around a line parallel to the V-axis through a point that lies at -0.5, 0, 1.

Before moving to the next section, clear all the data using:

FILE ↓
Close

Manipulating *.opc* and *.sat* files

The standard file extension for a data file prepared in the OPERA-3d modeller is a *.opc* file. In order to save a model which has just been prepared in this file format simply select

File → **Save as new model data**

and enter a filename. You will notice that in the File Type Filter only *.opc* files are available.

The format of the standard OPERA-3d file extension (*.opc*) is very similar to the *.sat* file format. The *.opc* file will, however, contain some extra information, not normally found in a *.sat* file, relating to conductor data.


In order to save the data as a *.sat* file, the following procedure is to be adhered to:

1. Ensure that the **Pick Entity** icon is enabled.
2. Click on the **Pick Bodies** icon.
3. Double click on the entities you wish to save.
4. Select
 Operations → **Export picked bodies** → **SAT File**
 and enter an appropriate name in the dialogue box.

Note that certain information on the model specific to OPERA-3d, such as conductor data, may be lost.

A *.sat* file can also be loaded and manipulated such that an OPERA-3d database can be created from this file. Choose

File → **Open**

or use the file open icon .

In the dialogue box, select **SAT files (.sat)** from the file type filter. All *.sat* files available in this directory should appear. Select the *.sat* file you wish to manipulate and subsequently click on **open**.

Mesh Control

Mesh control is an important feature of the Solid Modeller, as it allows the user to refine a Finite Element Mesh in regions where fields are rapidly changing or where greater accuracy is required. Mesh control is, in addition, critical to the mesh generation process: poor mesh settings can lead to mesh generator failures.

The model of a simple C-cored electromagnet will be used to demonstrate some mesh control features. Figure 5.1 illustrates the model, which constitutes of 4 cells, namely the **iron** cell, the **airgap** cell (air between the pole pieces), the **airin** cell (air space inside the C-core) and finally the background (surrounding) air (not shown).

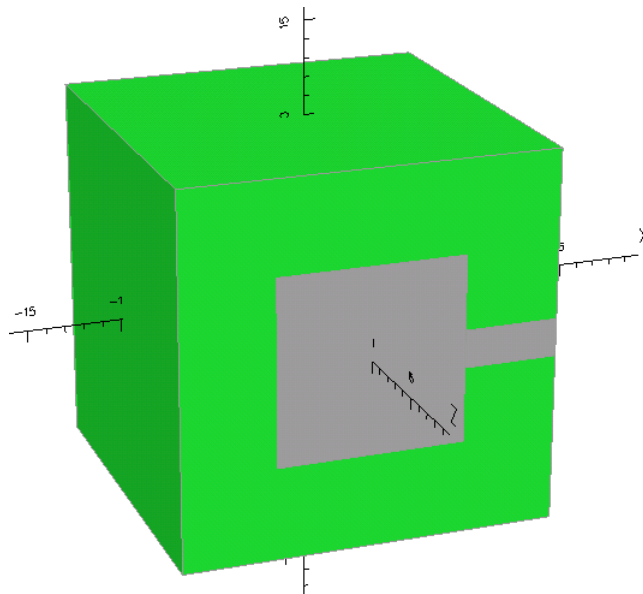




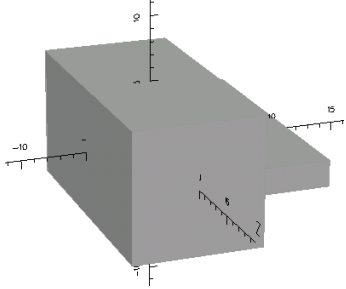

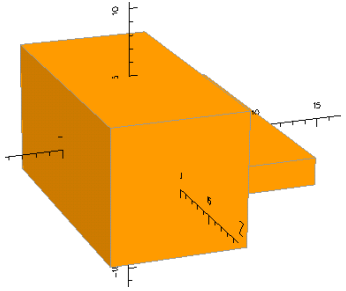


Figure 5.1 A C-core electromagnet (surrounding air omitted)

Building the model


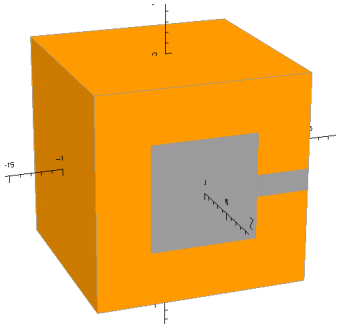
The following sequence is required to build the model above. It is assumed that the previous sections have been followed, so that full details are not required here. Create 3 blocks with the following dimensions:

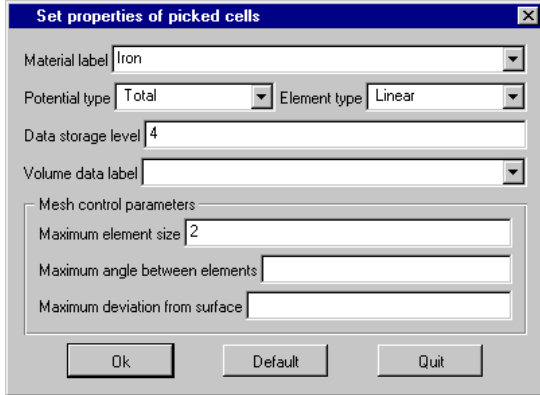
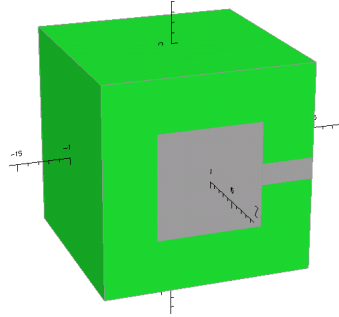
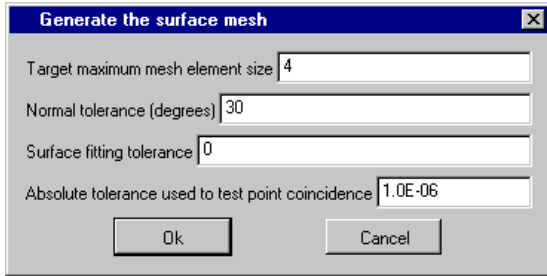
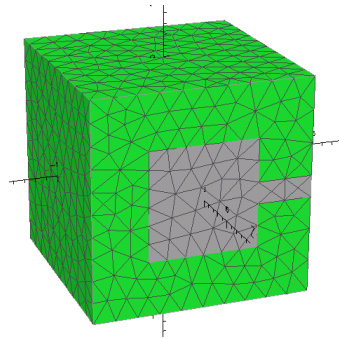
Name	First Corner	Opposite Corner
Block 1	-10,-10,-10	10,10,10
Block 2	-5,-5,-10	5,5,10
Block 3	5,-1,-10	10,1,10

All three block are combined. However since **Block 1** obscures the other two, the following sequence is recommended.

Select Hide Entity followed by Pick Bodies icons  	Select Block 1 in order to hide it initially.	
Select Pick Entity icon 	Select Block 2 and Block 3 .	
Select Unhide Entities icon 	This unhides Block 1 .	
Select Pick Bodies icon 	Select Block 1 , noting that Block 2 and Block 3 remain selected. All three are in this way selected, even though Block 1 is obscuring the other two.	
Operations → Combine bodies → Union without regularisation		


The C-core is now defined to be label **Iron**, with a total potential.

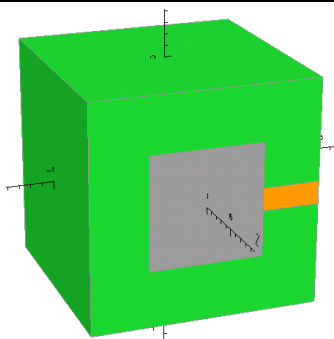
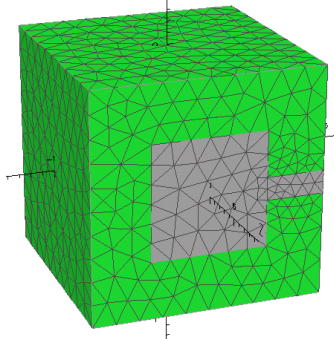
Select Pick Cells icon 	Select Block 1 . Note that there is only a single body, but 3 cells. The cells can be assigned properties separately.	
----------------------------------------------------------------------------------------------------------------------	------------------------------------------------------------------------------------------------------------------------------	---------------------------------------------------------------------------------------

Properties → Cell properties		
	<p>Complete the parameter box as follows:</p> 	
Model → Create model body Model → Generate surface mesh		
	<p>Follow this sequence to create the surface mesh. Complete the parameter box to set maximum mesh size to 4.</p> 	


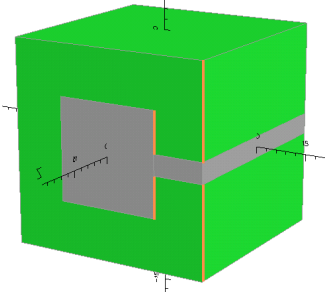
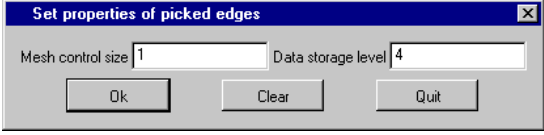
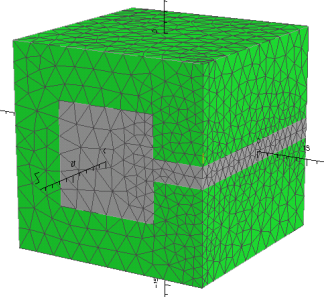
Modifying the Mesh Properties

The mesh needs to be fine in the iron (especially near the airgap) and in the airgap itself, where most of the MMF is dropped. In order to refine the mesh in the entire iron region the user would have to follow the sequence below.


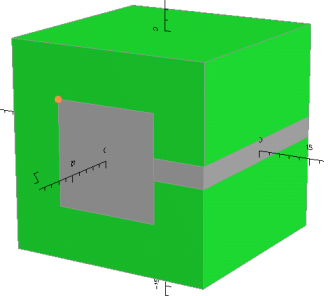
Ensure the **Pick Cells**  icon is enabled. To make changes to the model, it is first necessary to delete the model body (note this does not delete the component parts of the model).

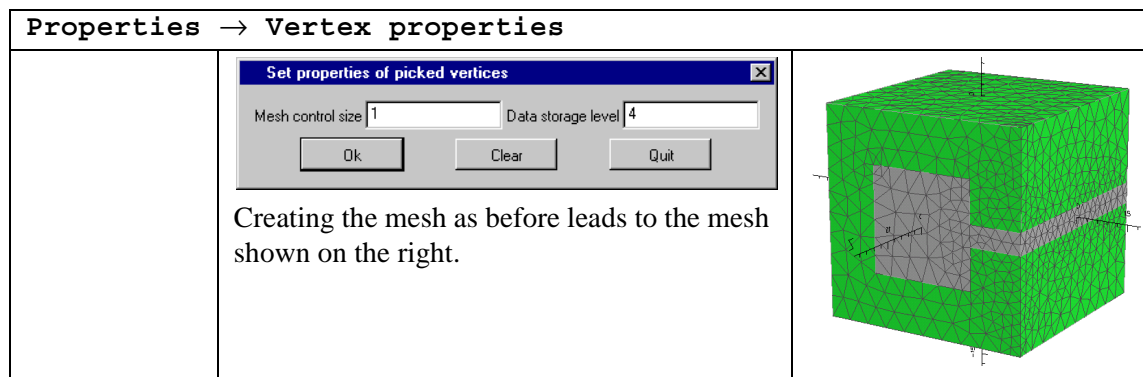
Model → Delete model body		
	Select Block 3 , the air gap.	
Properties → Cell properties		
	Set Data Storage Level to 5, and Maximum element size to 1.	
	Select Block 2 , the central air gap, and set in the cell properties the Data Storage Level to 3, Maximum element size to 3.	
Model → Create model body Model → Generate surface mesh		
	Complete the parameter box as follows when creating the surface mesh: <div data-bbox="461 1129 1005 1400"> <p>Generate the surface mesh [X]</p> <p>Target maximum mesh element size <input type="text" value="4"/></p> <p>Normal tolerance (degrees) <input type="text" value="30"/></p> <p>Surface fitting tolerance <input type="text" value="0"/></p> <p>Absolute tolerance used to test point coincidence <input type="text" value="1.0E-06"/></p> <p><input type="button" value="Ok"/> <input type="button" value="Cancel"/></p> </div>	

To further refine the mesh, some edges will be selected, and a finer mesh defined.

Model → Delete model body		
Select Pick Edges icon 	Select the edges shown in the figure on the right.	
Properties → Edge properties		
	 <p>Creating the mesh as before leads to the mesh shown on the right.</p>	

Mesh control can also be used on faces of a cell, and vertices. The former is useful if the mesh over the whole face needs to be refined. To refine around a specific vertex, the following sequence can be used:

Select Pick Vertex icon 	Pick the vertex indicated in the figure on the right.	
-----------------------------------------------------------------------------------------------------------------------	-------------------------------------------------------	---------------------------------------------------------------------------------------



Note that the data storage level helps to avoid ambiguities in the properties of edges, faces etc. when the data is merged to form a **model body**. Data set in a greater storage level will always be kept.


Before moving to the next section, clear all the data using:

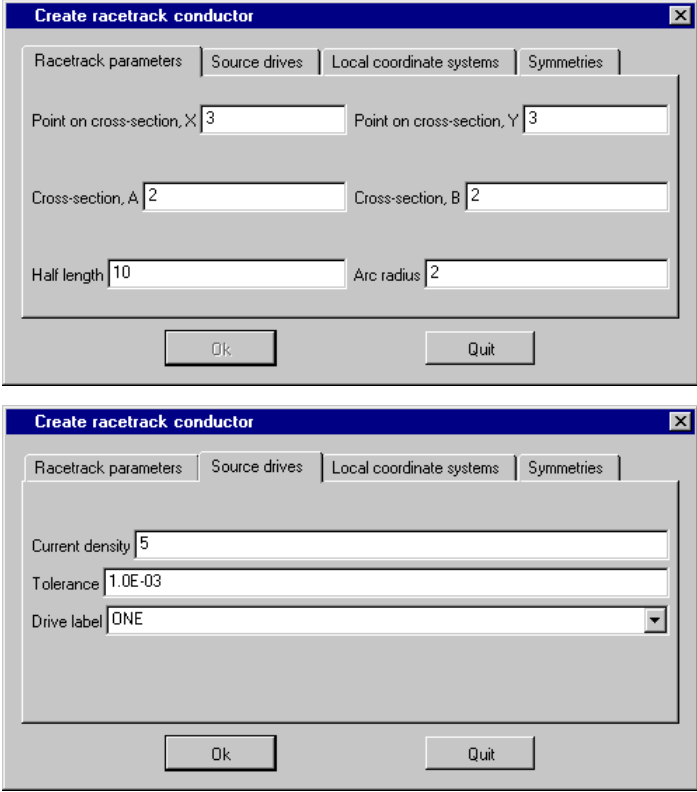
FILE ↓
Close

Defining and Editing Conductors

In OPERA-3d conductors do not normally form part of the Finite Element mesh. The modeller has a range of pre-defined (parameterised) conductor geometries, such as solenoids and racetracks, though simple (generic) shapes are also available and can be used to construct any conductor shape which is not already offered in parametric form.

Create a New Conductor

As an example, the racetrack conductor for the C-cored electromagnet introduced earlier can be defined by selecting the **Racetrack** icon:  and specifying the settings in the four tabs of the dialogue box. The four tabs are shown following, and they will produce the racetrack conductor of Figure 5.2.



Create racetrack conductor

Racetrack parameters | Source drives | Local coordinate systems | Symmetries

Point on cross-section, X: 3 Point on cross-section, Y: 3

Cross-section, A: 2 Cross-section, B: 2

Half length: 10 Arc radius: 2

Ok Quit

Create racetrack conductor

Racetrack parameters | Source drives | Local coordinate systems | Symmetries

Current density: 5

Tolerance: 1.0E-03

Drive label: ONE

Ok Quit

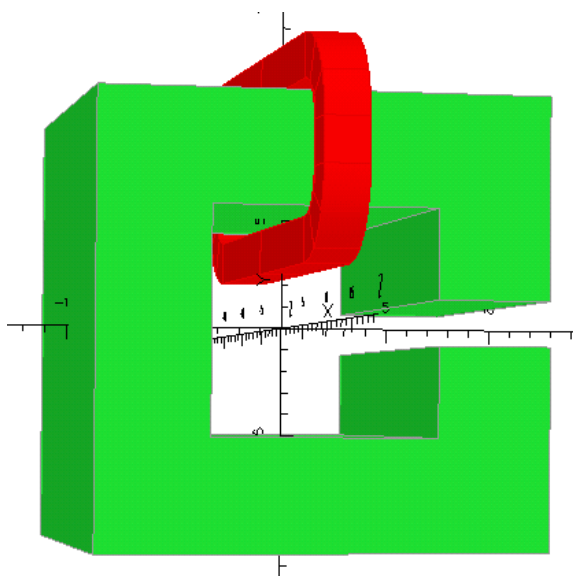
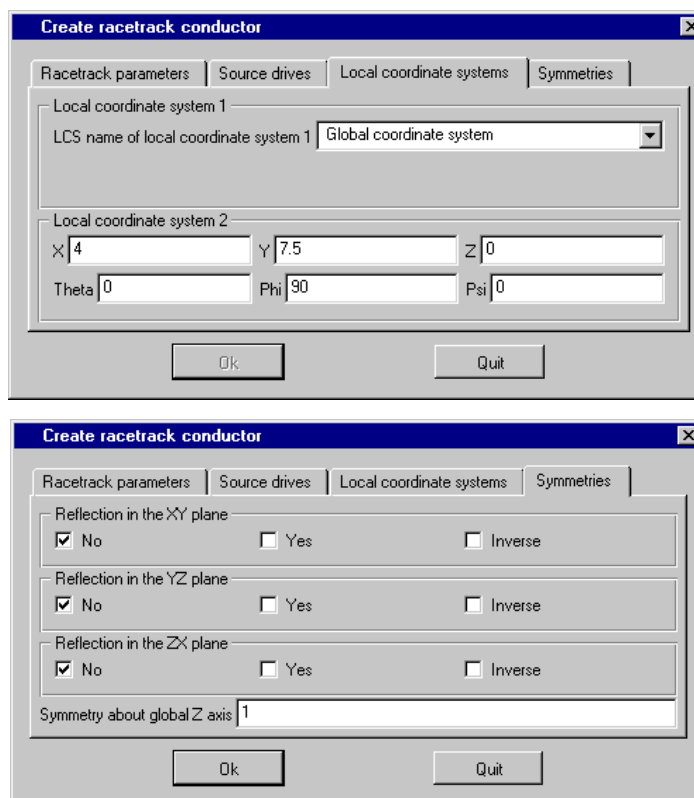



Figure 5.2 The racetrack conductor.


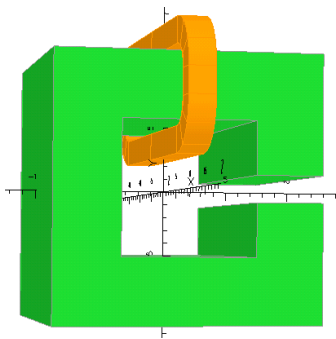
Save the Model

At this point it is recommended to save the C-core and the conductor in a file for use in a later example, using **File** → **Save as new model data....** and give a suitable file name.

Modify an Existing Conductor

In order to modify an existing conductor, the user would have to carry out the following steps.

Ensure the **Pick Entity**  icon is enabled

Press the Pick Conductors icon 	Double-click on the conductor to select it. In the figure on the right, the air regions were first hidden, but this is not necessary.	
Operations → Modify Conductors → Racetrack		
	Activate and edit the Modify Conductors Menu , which is similar to the dialogue box of page 5-40 .	

Uses of Volume Data

Volume data is useful for setting the following attributes:

- Permanent magnet directions,
- Anisotropy,
- Current density assignment (for ELEKTRA-SS/TR),
- Velocity (for ELEKTRA-VL).

The use of the volume data to assign anisotropic properties to the iron of the C-core magnet previously discussed will next be demonstrated. It will also be necessary to define a Background region, extending from (-30,-30,-30) to (30,30,30) in order to analyse the model. Alternatively, the data file (.opc) is provided with the OPERA installation (see the sub-folder *Examples/3D*) if required.

Figure 5.3 illustrates the **B**-field distribution in the C-core electromagnet. The model was solved with non-linear magnetic properties for the iron (a linear relative magnetic permeability of 500 was also entered in the material properties purely for the sake of completeness: the value will only be used if linear properties are chosen instead).

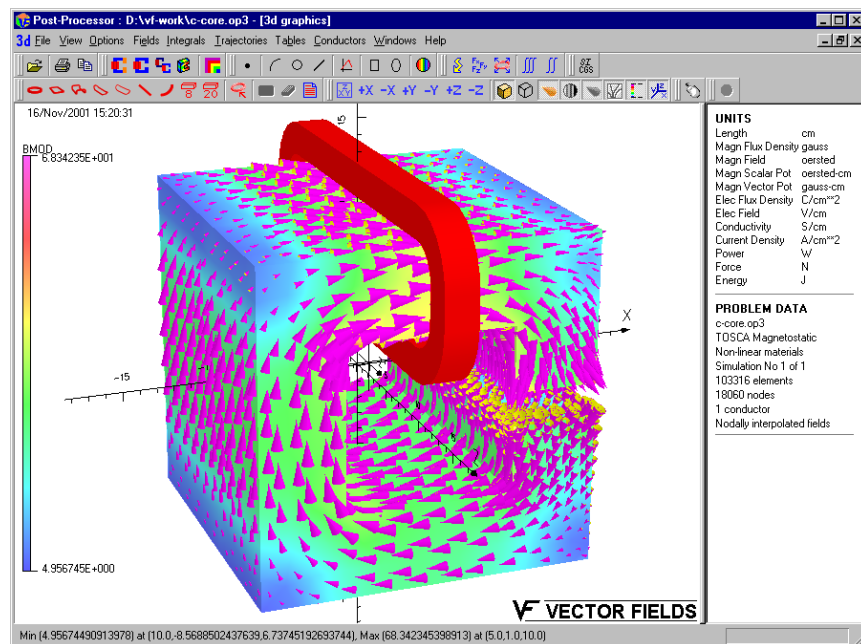


Figure 5.3 B-field Distribution in the electromagnet.

Figure 5.4 illustrates the flux density distribution taken along a line in the iron, just inside the racetrack coil. It can be observed that, due to the end winding, the flux density is higher near the edges of the iron, though the increase in flux density from the centre to the edge is smooth. This is also a natural effect, and is due to the fact that end winding induced field can travel axially into the iron which is not laminated.

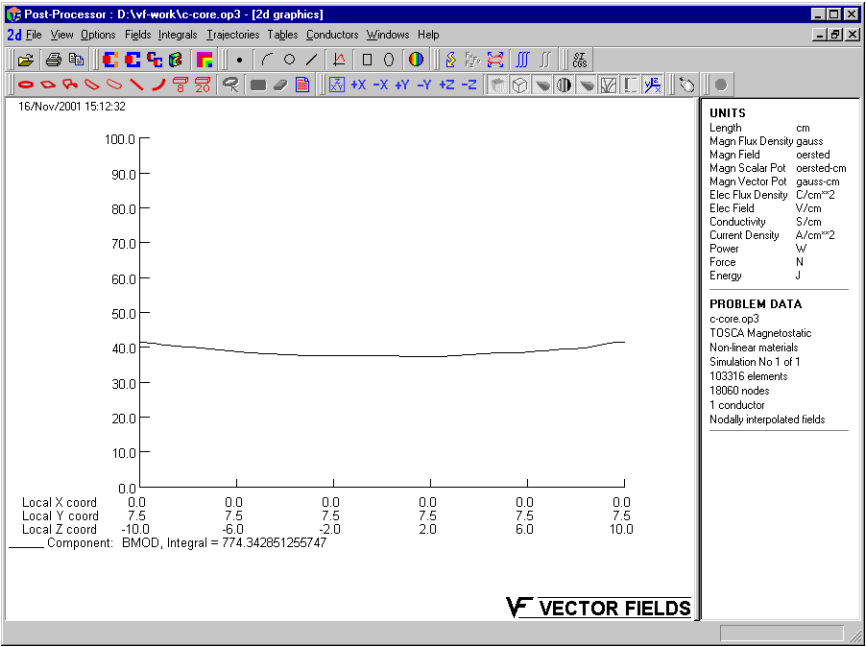



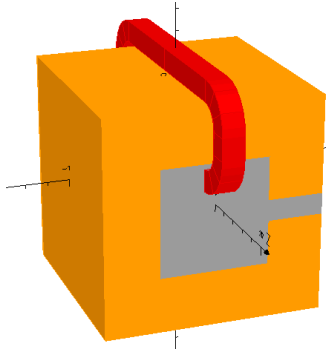
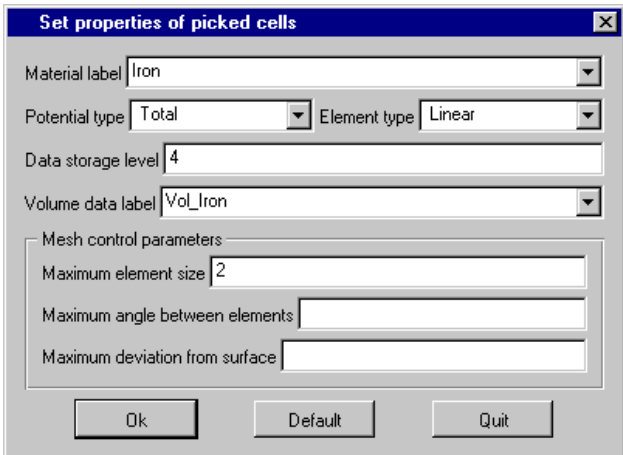
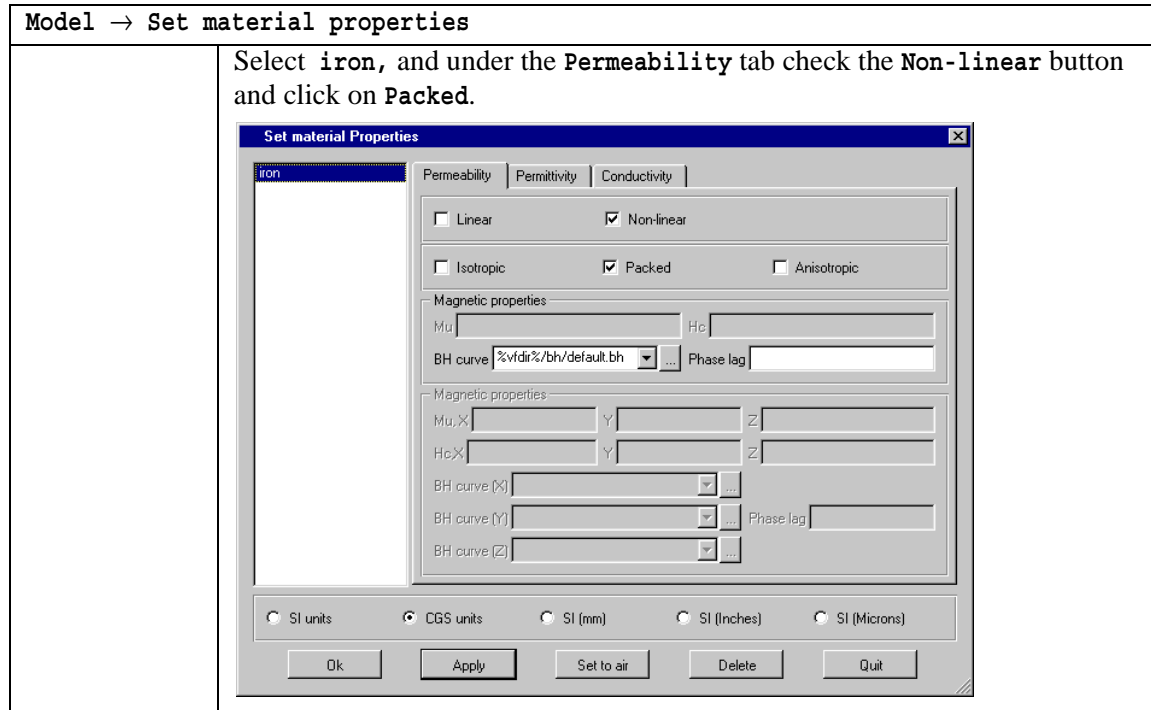


Figure 5.4 Flux density in the iron

In order to solve the same model but now with anisotropy, the user should carry out the following steps (note that the steps would be similar for any model).

Select the Hide Entity icon	Double click on the Background region, if it exists, to hide it.	
		

<p>Select the Pick Entity icon</p>  <p>and Pick Cells</p> 	<p>Double click on the iron</p>	
<p>Properties → Cell properties</p>		
	<p>In the Cell Properties table enter a Volume Data Label (say Vol_Iron)</p> 	



The default BH curve is available, and can be accessed by including the variable **%vmdir%** (for Windows operating system) or **\$vmdir** (for UNIX operating systems) in the pathname.

To set the volume properties:

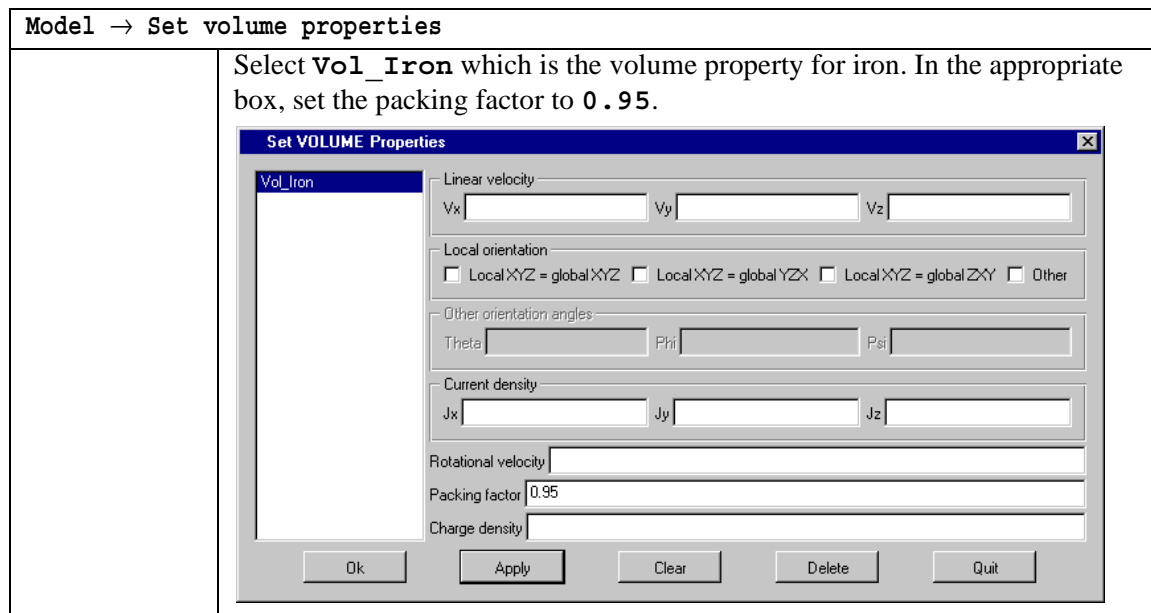


Figure 5.5 illustrates the result of a line graph of flux density in the anisotropic iron (just inside the coil). End effects are now more pronounced at the edges of the iron, as the \mathbf{B} -field can no longer penetrate in the axial direction. In physical terms this would imply that the final few laminations of the core will operate at higher flux densities.

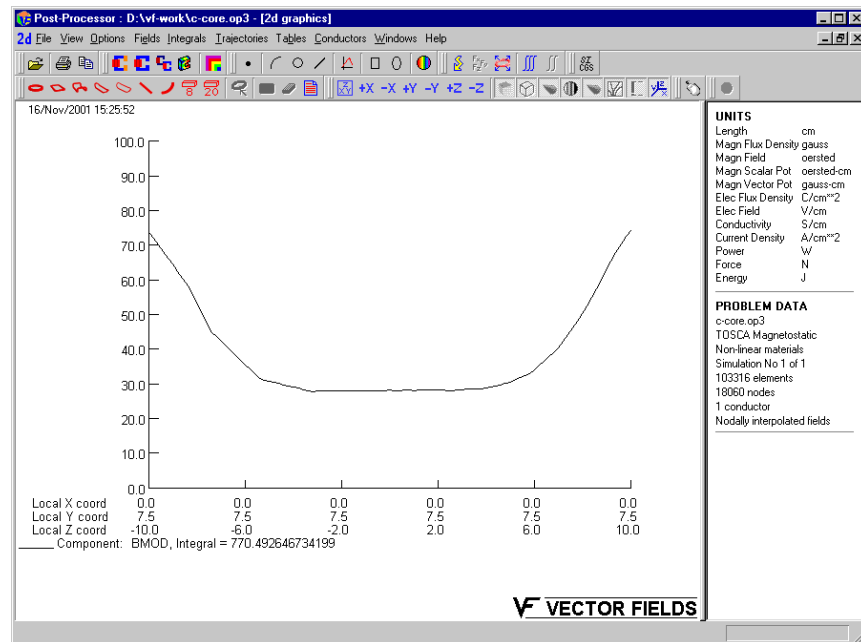
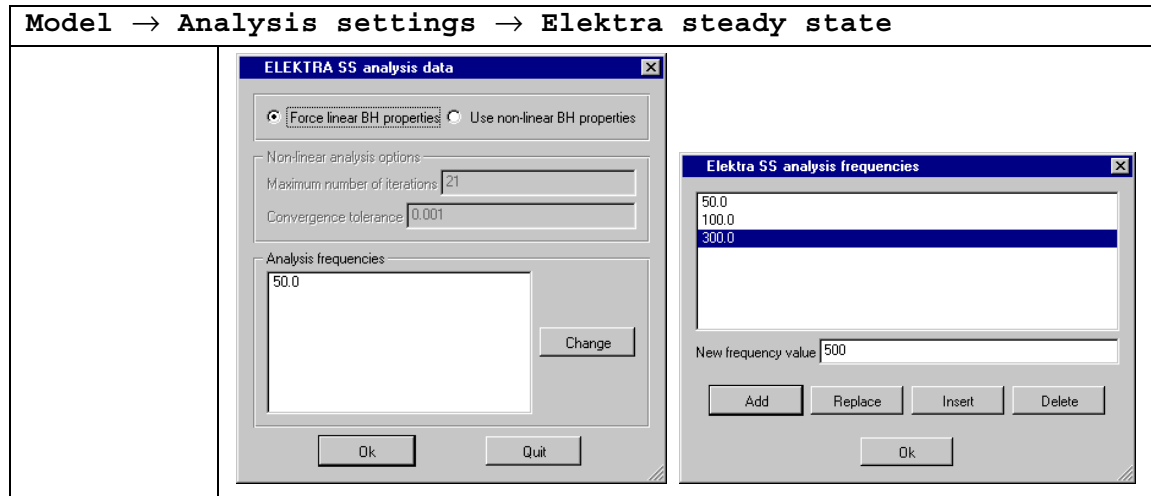


Figure 5.5 Flux density in the anisotropic C-core

Creating .op3 files

The C-core electromagnet, shown in [Figure 5.1, on page 5-34](#), can be driven at multiple frequencies. In order to create a single database, containing solutions to multiple frequencies, the user should select



In the **Analysis data** window the **change** button can be used to change (or add to) the default 50Hz frequency simulation. New values can be typed in, and the entry confirmed with the **Add** button. Entries can be deleted by clicking on them and subsequently selecting the **Delete** button.

In the window called **Create simulation for analysis**, the **Analysis module** (solver to be used), **Units**, **Element** and **Surface Element Type**, **Filename** and **Simulation Title** are entered. The title of the simulation is entered in the area provided under the database filename box. Surfaces can be assigned with a different type of element to the volume they enclose. A selection of **Mixed Elements** adopts the default settings, as entered in the **Cell Data** and **Face Data** windows.

To create a database (.op3) file, complete the parameter box after selecting:

Model → Create Analysis database

Create simulation for analysis

☒ Create new database file ☐ Add simulation to existing database

Analysis module

☐ Tosca magnetic ☐ Tosca electrostatic ☐ Tosca current flow ☐ Scala

☒ Elektra SS ☐ Elektra VL ☐ Elektra VL (rotational) ☐ Elektra TR

☐ Soprano SS ☐ Soprano EV

Units

☐ SI ☒ CGS ☐ mm ☐ Microns ☐ Inches

Element type

☐ Linear ☐ Quadratic ☒ Mixed

Surface element type

☐ Linear ☐ Quadratic ☐ Mixed ☒ Curved

Database: Filename

Title of Problem

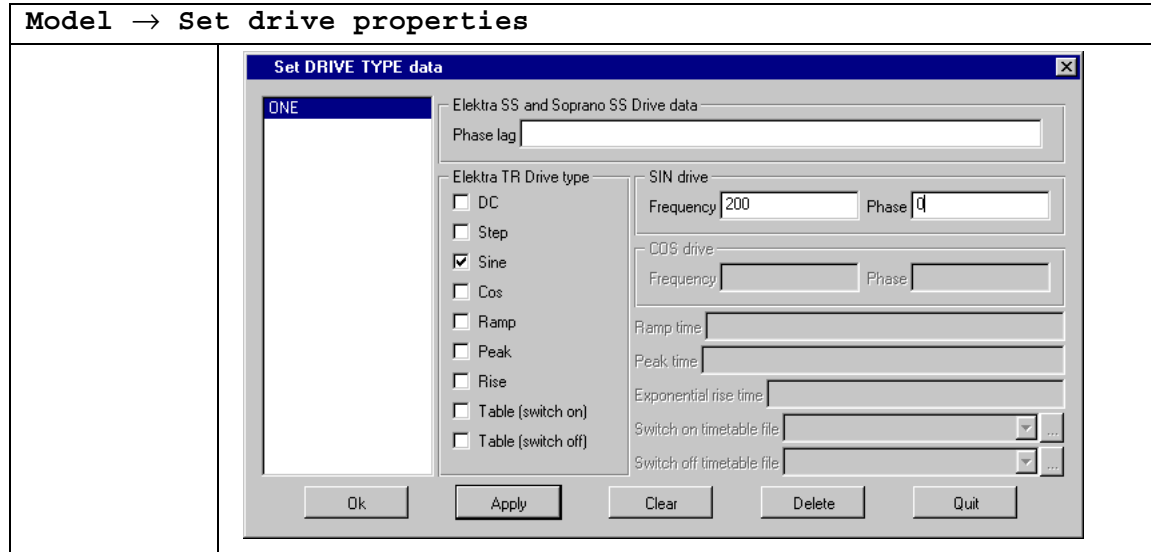
Ok Cancel

The same model can also be used to examine the transient behaviour of the electromagnet. OPERA-3d can be used to estimate how the flux builds up in the C-core when a transient is applied to the exciting coil. Amongst others the following functions are available:

- Step
- Ramp
- Sine
- Cosine
- User-defined Table

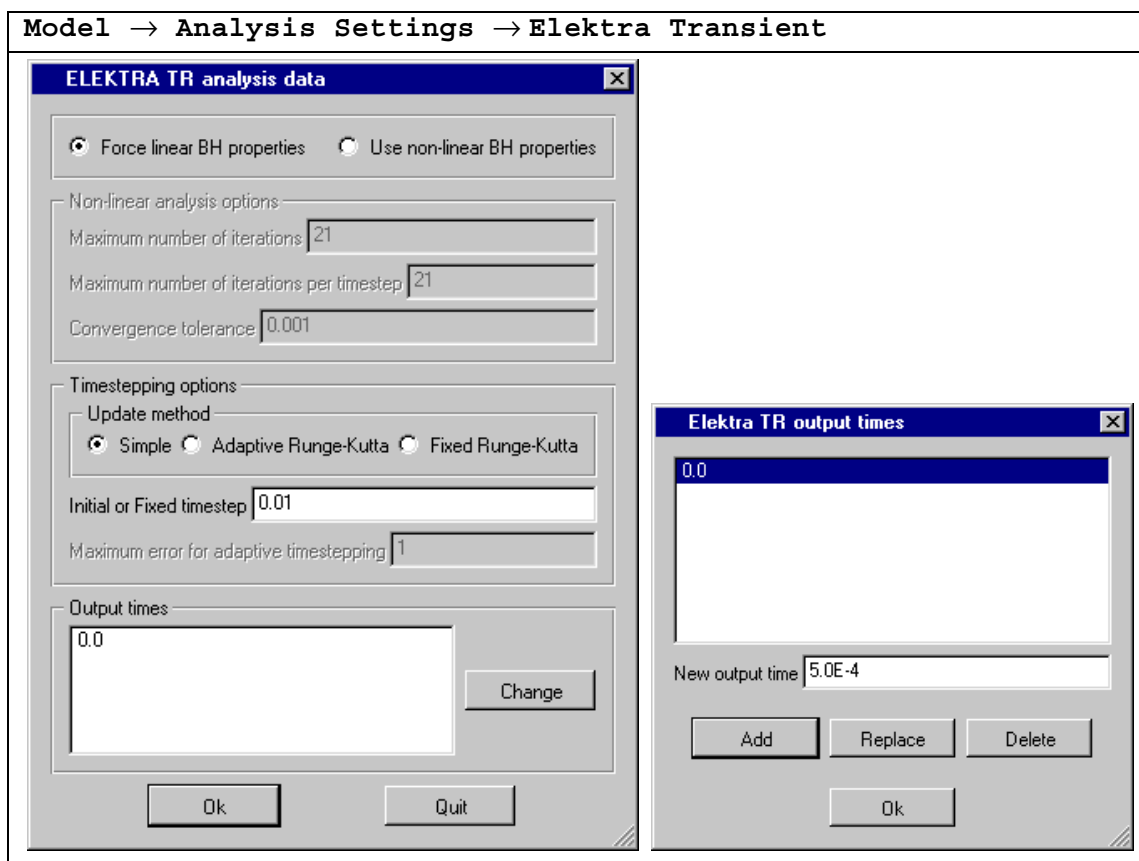
The latter option can be used to define the shape of the function to be applied to the exciting coil, if none of the pre-defined functions available are appropriate.

The drive information is used by the OPERA-3d Transient solvers (ELEKTRA-TR and CARMEN) to define the type of drive to be applied to each label. Drive information can be accessed via



Drive labels, each corresponding to one (or set of) conductor(s), must be individually selected and assigned the appropriate properties. Note that different entries are relevant to the variety of drives which are available. In this instance, a sinusoidal source has been selected and therefore the **Frequency** and **Phase** entries must be filled in.

In order to set up the transient solution the user should activate the **ELEKTRA-TR Analysis** menu shown below. The preferred output times can be set by selecting the **Change** button which will activate the **Output Times** dialogue box.



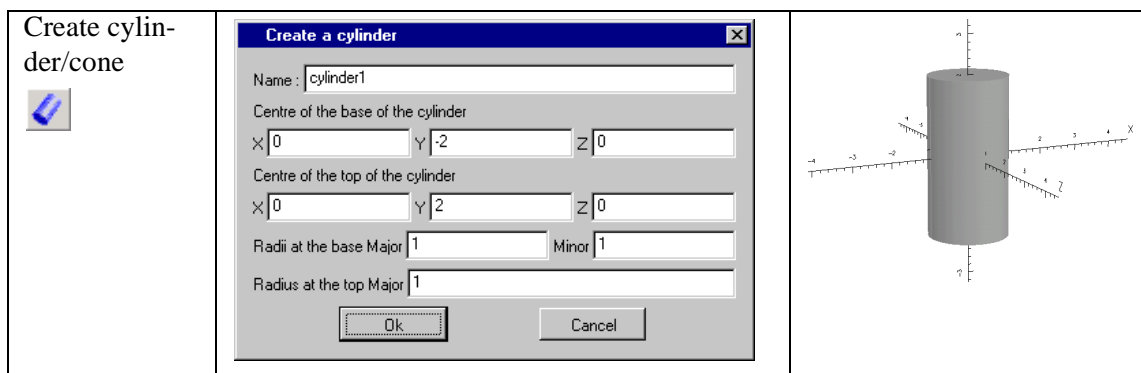
Before moving to the next section, clear all the data using:

FILE ↓
Close


Local Coordinate Systems

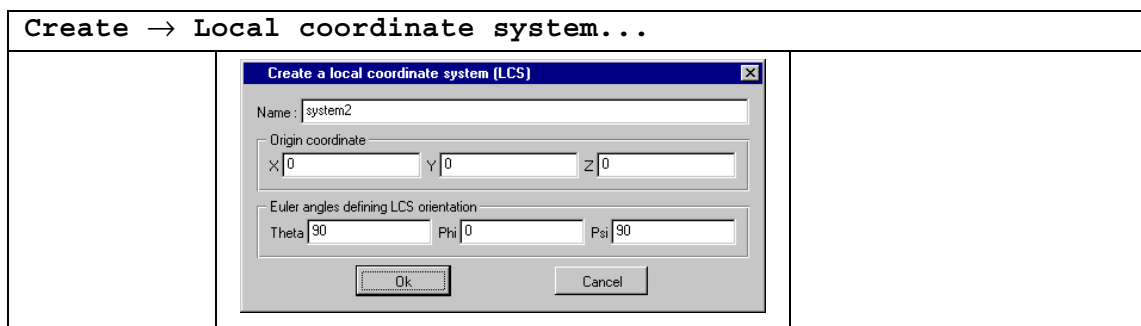
Local coordinate systems can be used to ease the definition of the Finite Element Model. For instance, should the definition of the same object be needed repeatedly, but at different positions, the body local coordinate system could be changed in each instance, instead of its definition parameters.

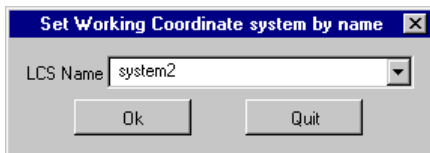

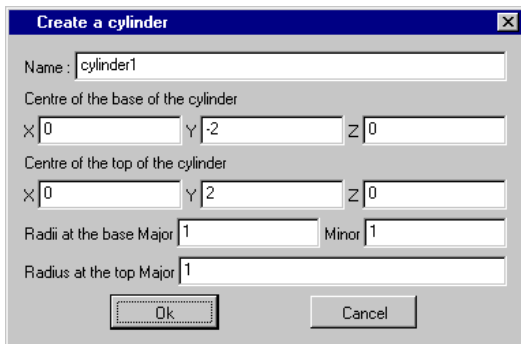
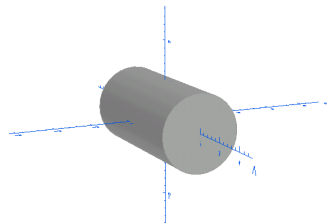
A cylinder can be defined on the global coordinate system as follows:




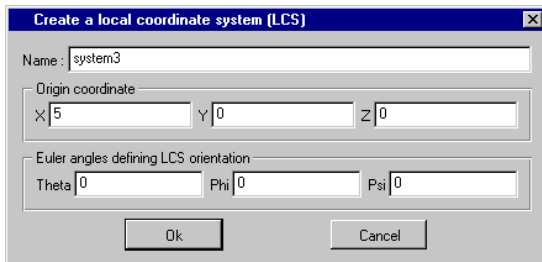
A dimensionally identical cylinder but with its axis parallel to the Z-axis, can be defined by first creating a local coordinate system and subsequently defining the same cylinder on this new local system.

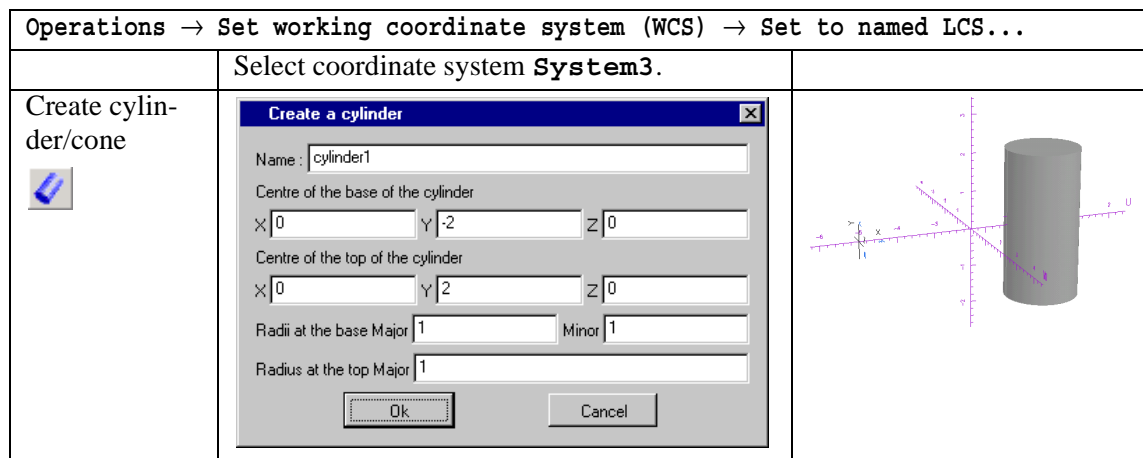
Use **Undo** icon  to remove the cylinder.



Operations → Set working coordinate system (WCS) → Set to named LCS...		
	<p>The previously defined system is available to select:</p>  <p>The dialog box 'Set Working Coordinate system by name' has a title bar with a close button. It contains a text field 'LCS Name' with 'system2' selected, and 'Ok' and 'Quit' buttons at the bottom.</p>	
<p>Create cylinder/cone</p> 	 <p>The dialog box 'Create a cylinder' has a title bar with a close button. It contains the following fields: 'Name' (cylinder1), 'Centre of the base of the cylinder' (X: 0, Y: -2, Z: 0), 'Centre of the top of the cylinder' (X: 0, Y: 2, Z: 0), 'Radii at the base Major' (1) and 'Minor' (1), and 'Radius at the top Major' (1). It has 'Ok' and 'Cancel' buttons at the bottom.</p>	 <p>A 3D model of a cylinder is shown in a coordinate system. The cylinder is centered at the origin (0,0,0) and has a radius of 1. The axes are labeled X, Y, and Z.</p>

A cylinder can also be defined to be displaced from the original cylinder, in any direction. The following settings define a cylinder displaced in the positive X direction. As before, use the **Undo** icon  to remove the cylinder created above.

Create → Local coordinate system...		
	 <p>The dialog box 'Create a local coordinate system (LCS)' has a title bar with a close button. It contains the following fields: 'Name' (system3), 'Origin coordinate' (X: 5, Y: 0, Z: 0), and 'Euler angles defining LCS orientation' (Theta: 0, Phi: 0, Psi: 0). It has 'Ok' and 'Cancel' buttons at the bottom.</p>	




Building a Library of Sub-Models

The Modeller allows users to build a model from several parts, which can be created and saved separately. This example uses the C-core magnet, which has been created in the section about [Mesh Control on page 36](#) and saved in the section about [Defining and Editing Conductors on page 42](#). To illustrate how a model can be created from parts, this model will be loaded in twice.

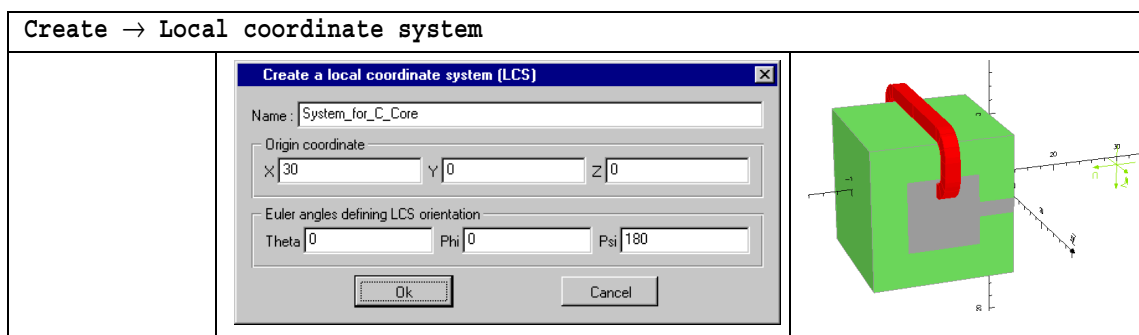
The geometry of a model or parts of a model can be saved in two different types of files (.opc file or .sat file). The main differences are that a .sat file cannot hold any conductor, magnetic material, boundary condition or volume data. The .sat files are the format of geometric model that allows data exchange between ACIS based CAD systems. The .opc files are Vector Fields' extension to .sat to support the OPERA analysis programs. Either type of file can be used to build up a model from parts.

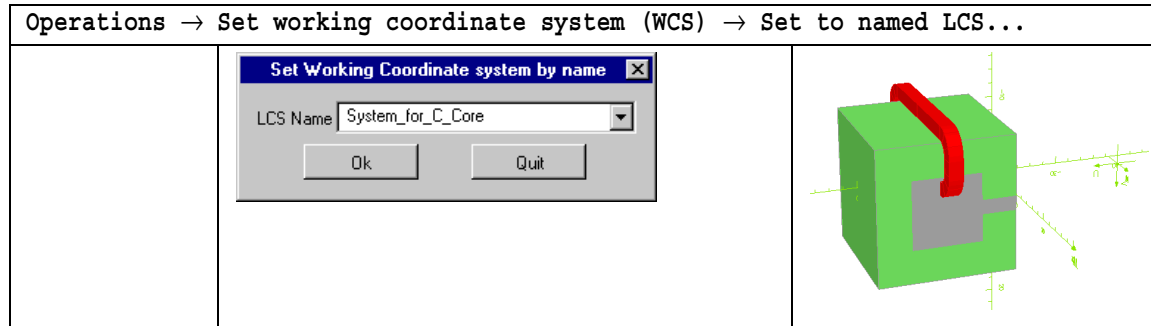
Before starting this example, the Modeller should be reset to its initial state using **File → Close**.

Load the C-core example from the previous sections into the Modeller, using the

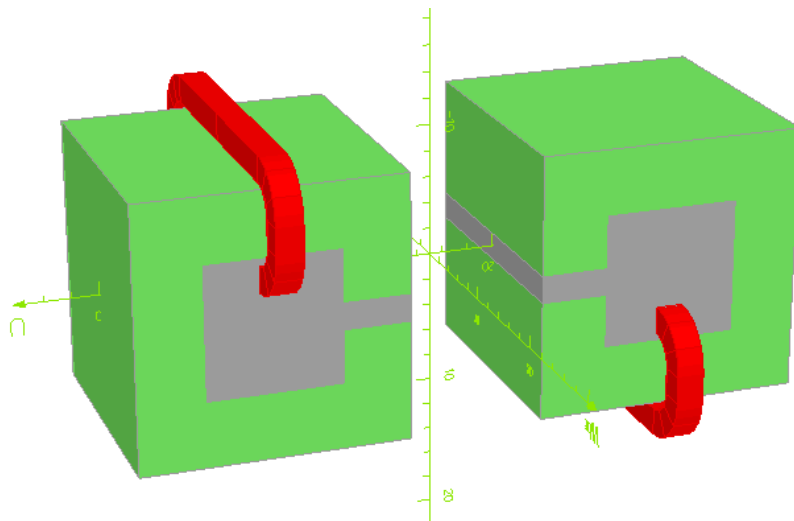
File Open icon:  and selecting the file from the appropriate directory. Use **Model → Delete Model Body**, and then delete the Background region. A data file (.opc) is provided in the OPERA installation (see the sub-folder *Examples/3D*) with the background region omitted.

Following this, the same C-core will be loaded a second time into the Modeller. A new local coordinate system will be used to displace the second copy and to rotate it by 180 degrees at the same time.





The C-core is again loaded into the Modeller, using: **Create** → **Insert from file...** and selecting the appropriate file.




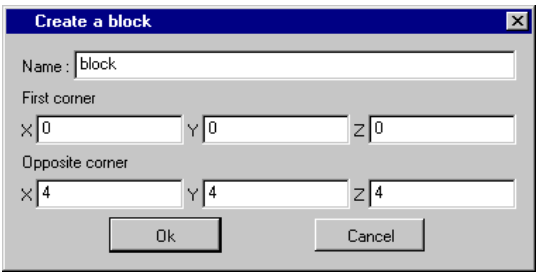
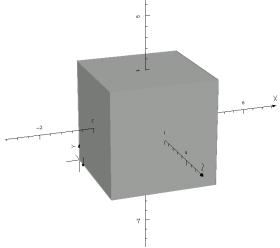


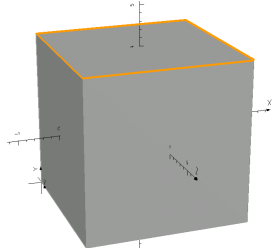
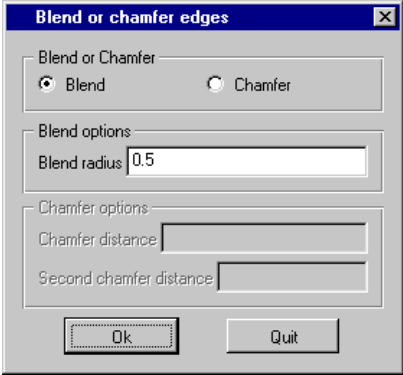
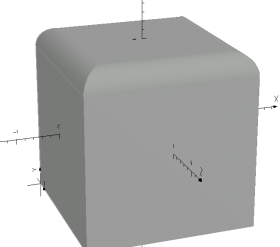
Based on this procedure, very complex models can be built up using parts from a user defined library.

Blend and Chamfer Edges


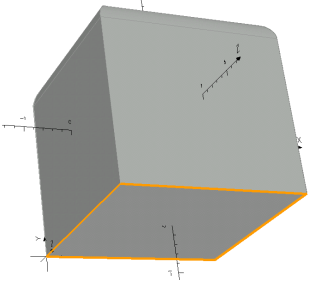
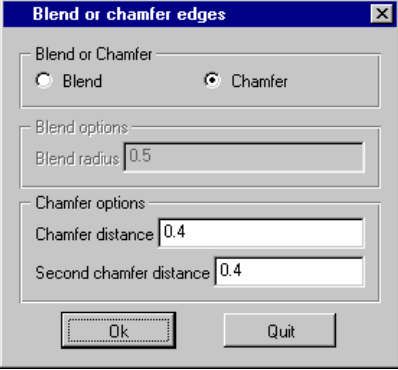
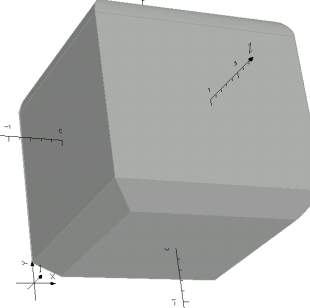
This example is self contained and needs no previous models. Therefore the Modeller should be reset to its initial state using **File** → **Close**.

Blend and **Chamfer** are both operations which can be applied to edges of a volume. The **Blend** operation smooths an edge using a radius, whereas the **Chamfer** operation smooths an edge by using a plane.

If the operations are applied to an “outer” edge, material will be cut away; if there is an “inner” edge, material will filled in. This gives a total number of 4 combinations, which all will be explained in this example.


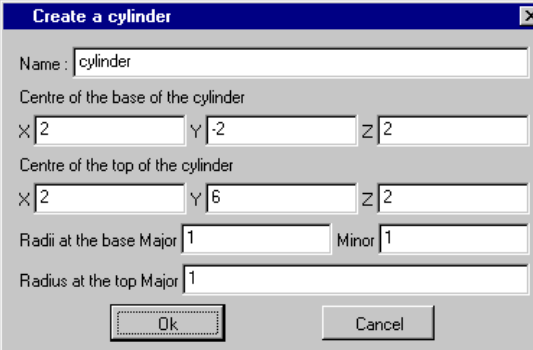
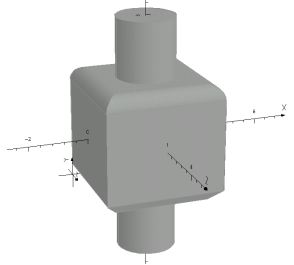
<p>Create a block</p> 		
<p>Pick edges and Pick entity</p>  	<p>Zoom in using the mouse and select the 4 edges on top of the block. The selected edges will turn orange.</p>	
<p>Operations → Blend or chamfer edges...</p>		
		


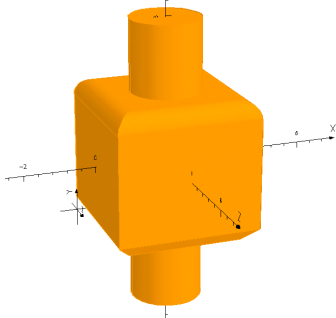

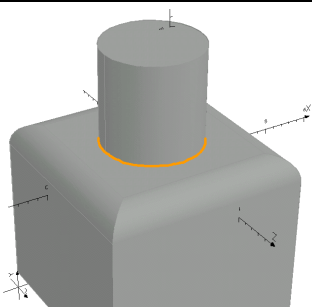
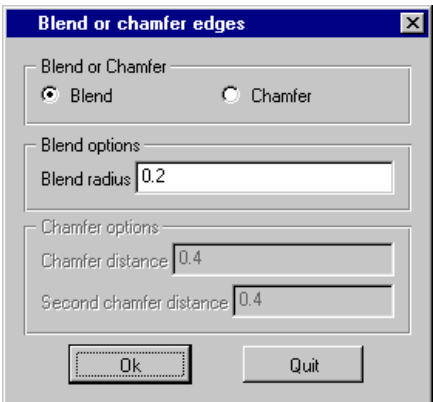
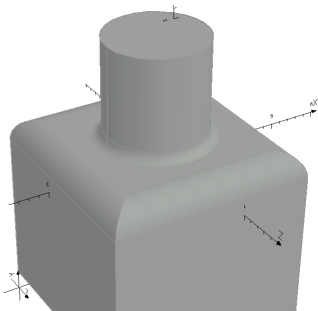
In the next operation the bottom of the block will be chamfered.

<p>Pick edges and Pick entity</p> 	<p>Rotate the block using the mouse, and select the 4 edges at the bottom of the block. The selected edges will turn orange.</p>	
<p>Operations → Blend or chamfer edges...</p>		
		


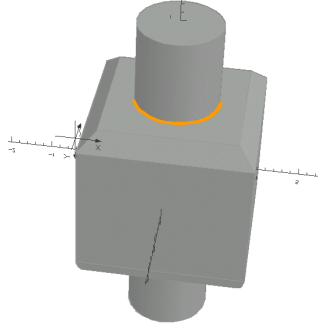
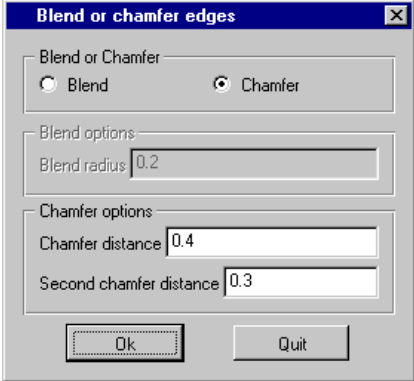
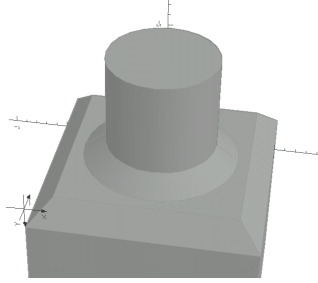
The second **chamfer distance** in the example above has been chosen to be the same as the first one, which will cut a 45 degrees section off the block. It is also possible to specify different values.

In the second half of this section a new cylinder will be defined, which cuts through the block. This allows **blend** and **chamfer** at an “inner” edge.


<p>Create a cylinder</p> 		
--------------------------------------------------------------------------------------------------------------	--------------------------------------------------------------------------------------	---------------------------------------------------------------------------------------

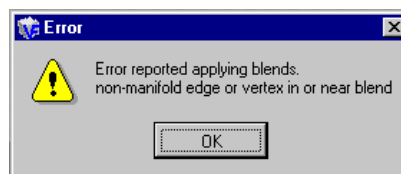
<p>Pick body and Pick entity</p> 	<p>Pick the cylinder and the block using the mouse. The sequence is not important as the next operation will be a Union with regularisation.</p> <p>Note that it is necessary to choose Union with regularisation so that only a single cell is created, and to ensure all edges have no more than two faces touching them (the edges are not manifold).</p>	
Operations → Combine bodies → Union, with regularisation		
<p>Pick edge</p> 	<p>Select the edge on top of the cube, where the cylinder meets the cube, as indicated on the right picture.</p> <p>It does not matter whether a 180 degrees section or both 180 degrees sections are selected; the program will always perform the operation on the whole circle (a feature of the underlying ACIS kernel).</p>	
Operations → Blend or chamfer edges...		
		

The last operation will be a **Chamfer** of the bottom junction. Use the mouse to rotate the body and select the edge where the cylinder meets the cube.

<p>Pick edge</p> 	<p>Select the edge at the bottom of the cube, where the cylinder meets the cube, as indicated on the right picture.</p>	
<p>Operations → Blend or chamfer edges...</p>		
		


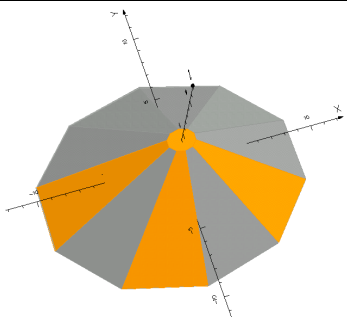
The **Blend** and **Chamfer** operations can only be applied if an edge is not manifold i.e. where more than two faces meet at an edge.

This can be demonstrated by using **Undo**  to return to the Boolean **Union** on [page 5-59](#) and making the **Union**, without **regularisation**. This will create 4 cells in the body and the edge at the junction of the cylinder and block will now belong to four faces. When the chamfer or blend is applied the user will receive a warning message.


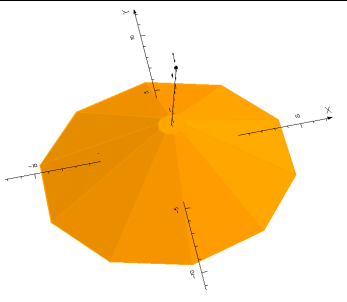




Swift Selection of Entities

Reset the Modeller to its initial state using **File** → **Close**.

<p>Create a pyramid</p> 	<div><div>Create a prism / pyramid</div><div>Name : <input type="text" value="pyramid"/></div><div>Number of sides <input type="text" value="10"/> Height <input type="text" value="8"/></div><div>Major radius at base <input type="text" value="10"/> Minor radius at base <input type="text" value="8"/></div><div>Major radius at top <input type="text" value="1"/></div><div><input type="button" value="Ok"/> <input type="button" value="Cancel"/></div></div>	
-----------------------------------------------------------------------------------------------------------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-------------------------------------------------------------------------------------

This pyramid has many faces, and selecting them one by one needs 12 mouse double-clicks. This can be reduced to 1 mouse double-click and a simple operation by the following procedure.

<p>Pick body and Pick entity</p> 	<p>Select the pyramid as a body.</p>	
<p>Pick face</p> 	<p>Press the Pick face button. The pyramid is still selected as a body.</p> <div><div>Component</div><div>1(1)</div><div>1(0)</div><div>12(0)</div><div>30(0)</div><div>20(0)</div><div>0</div><div>Global coordinate system</div></div>	
<p>Change type of picked entities</p> 	<p>Press the Change type of picked entities button. As a result of this operation all faces of the selected body are selected instead. This can be seen on the display of the entities, which are listed on the lower right hand side of the Modeller window.</p> <div><div>Component</div><div>1(0)</div><div>1(0)</div><div>12(12)</div><div>30(0)</div><div>20(0)</div><div>0</div><div>Global coordinate system</div></div>	

The procedure can be very useful for example in electrostatic models, where the faces of a body have to be selected to apply a label for voltage boundary conditions. Another application might be to select all edges of a body in one go and apply **Blend** or **Chamfer** operations.

Chapter 6

Analysis Programs

Introduction

This manual describes the electromagnetic field analysis programs within the OPERA-3d environment. The analysis programs incorporate state of the art algorithms for the calculation of electromagnetic fields and advanced finite element numerical analysis procedures. A brief introduction to the use of finite elements is given in section “[The Finite Element Method](#)” on page 6-2 as an aid to application engineers who need to understand the limitations of the technique and evaluate the validity of their results. In sections “[Codes of Practice](#)” on page 6-6 and “[Solution Errors](#)” on page 6-7, some further discussions on good codes of practice, and techniques for reducing errors are presented.

Finite element discretisation forms the basis of the methods used in the analysis programs. This widely applicable technique for the solution of partial differential equations requires special enhancements to make it applicable to electromagnetic field calculations. Access to these features is supported by the OPERA-3d pre processor which provides facilities for the creation of finite element models, specification of complicated conductor geometry, definition of material characteristics including for example, non-linearity and anisotropy and graphical displays for examination of the data. Full details are given in the OPERA-3d Reference Manual.

The Finite Element Method

The Finite Element method is used to obtain solutions to partial differential or integral equations that cannot be solved by analytic methods. Partial differential and integral equations describe the spatial and temporal variation of a field either directly in terms of the field variable, for example the magnetic flux density \mathbf{B} , but more often using a potential function that is related to the Field by a gradient (∇) or curl ($\nabla \times$) operation. The finite element method is generally applicable to any problem with any type of non-linearity. The method is based on division of the domain of the equation (volume of space in which the equation is satisfied) into small volumes (the finite elements). Within each finite element a simple polynomial is used to approximate the solution.

The concepts used in finite element analysis are independent of the number of space dimensions, however it is convenient to use a simple one space dimension problem in order to make the algebra straightforward and explanatory diagrams easy to view. Consider a Poisson type equation describing a potential function ϕ in one dimension:

$$\nabla \cdot \epsilon \nabla \phi = \rho \quad (6.1)$$

The potential function ϕ might be an electrostatic potential, in which case ρ would be a line charge density. In order to define ϕ , boundary conditions are required,

these may be either assigned values of ϕ or its derivative $\frac{\partial \phi}{\partial x}$, for example:

$$\frac{\partial \phi}{\partial x} = 0 \quad (6.2)$$

In all electromagnetic field examples it is essential that the potential is defined at one point in the domain at least, otherwise an infinite number of solutions could be generated by adding an arbitrary constant to the solution.

To solve equation 6.1 using a finite element method the domain is divided into line elements. A typical first order line element would have two nodes numbered e.g. 1 and 2. Within this element the potential ϕ will be approximated by a linear polynomial:

$$\phi(x) = a + bx \quad (6.3)$$

The electrostatic potential ϕ will be continuous over the domain, although its derivatives may be discontinuous if the permittivity ϵ changes discontinuously. The finite element model should be capable of representing this behaviour and it is therefore convenient to characterize the polynomial shown in equation 6.3 by

the values of ϕ at the nodes of the element and use the same nodal value to characterize the polynomials in other elements that meet at the node. A further simplification is introduced by rewriting equation 6.3 in terms of nodal shape functions N_i defined such that:

$$\begin{aligned} N_i(x) &= 1; & x &= x_i \\ N_i(x) &= 0; & x &= x_j, \quad j \neq i \end{aligned} \quad (6.4)$$

where x_i is the x coordinate of node i etc. The shape functions have the same polynomial form as the ϕ approximation, and equation 6.3 can be written as:

$$\phi(x) = N_1(x)\phi_1 + N_2(x)\phi_2 \quad (6.5)$$

The shape functions N_i are usually expressed in terms of a local coordinate system in the element. This can be used to simplify the expressions and furthermore avoids problems of numerical rounding errors. Using the local coordinate system ξ the shape functions can be written as:

$$\begin{aligned} N_1 &= \frac{1}{2}(1 - \xi) \\ N_2 &= \frac{1}{2}(1 + \xi) \\ -1 &\leq \xi \leq 1 \end{aligned} \quad (6.6)$$

The shape function for a particular node is only defined in the elements that use the node and is zero outside these elements. The approximation to ϕ is described as having local support when nodal shape functions of this type are used.

The discrete method of approximating the potential ϕ using characteristic nodal values and associated shape functions that determine the spatial variation of the approximation provides the basis on which several alternative procedures could be used to solve equation 6.1. Variational methods, least squares and weighted residual procedures are three of the most frequently used. Weighted residuals have wide application and they are used in the software to develop a numerical solution. An approximate solution ϕ is determined by requesting that this function should satisfy:

$$\int W(\nabla \cdot \epsilon \nabla \phi - \rho) dx = 0 \quad (6.7)$$

The weighted residual method can be used with either global (defined over the whole domain) approximations to ϕ or the local approximations discussed here. W is a weighting function from which the method gains its name. The Galerkin weighted residual method is the best choice for the types of equation arising in electromagnetism. In this case the basis functions approximating ϕ are also used

for the weights. Equation 6.7 is often referred to as a strong form because of the constraints it places on the functions that can be used in the approximation ϕ (the first derivative would clearly have to be continuous over the domain). In general a weak form of equation 6.7 is used to remove the derivative continuity requirement. This weak form is obtained by integrating equation 6.7 by parts (in more than one dimension this involves application of Green's theorem). Integrating equation 6.7 by parts to reduce the order of differentiation applied to ϕ gives:

$$\int_a^b (\nabla W \cdot \epsilon \nabla \phi + W \rho) dx - \left[W \epsilon \frac{\partial \phi}{\partial x} \right]_a^b \quad (6.8)$$

where a and b are the limits of the domain of the equation. The weak form has several advantages: the functions representing W and ϕ do not need derivative continuity and the natural boundary condition on the surface of the domain $\frac{\partial \phi}{\partial x}$ has emerged. Equation 6.8 leads directly to a numerical solution method, using the discrete finite elements and shape functions discussed above. Discretisation of the domain ab into line elements with their associated nodes gives a set of independent weighting functions (the shape functions of the nodes) from which a set of equations can be developed by requiring that equation 6.8 is satisfied independently for each weight function. The equation for weight function W_i , i.e. shape function N_i , is obtained from:

$$\sum_j \left(\int_a^b (\nabla N_i \cdot \epsilon \nabla N_j \phi_j + N_i \rho) dx \right) - \left[N_i \epsilon \frac{\partial \phi}{\partial x} \right]_a^b = 0 \quad (6.9)$$

for all elements containing node i . Taking all the equations for the different weight functions together gives a set of linear equations, which written in matrix form are:

$$\mathbf{K} \Phi = \mathbf{S} \quad (6.10)$$

where \mathbf{K} is a coefficient matrix (often called a stiffness matrix because of the background of finite elements in mechanics), Φ is a vector of unknown nodal potential values and \mathbf{S} the known right hand side vector derived from the prescribed line charge densities or assigned boundary conditions. An individual element of the stiffness matrix consists of terms of the form:

$$K_{ij} = \int_a^b \nabla N_i \cdot \epsilon \nabla N_j dx \quad (6.11)$$

Note that the local support of the shape functions means that although the integral in equation 6.11 is taken over the whole domain, only elements containing both

nodes i and j actually contribute. In the equations arising in electromagnetism the matrix equation 6.10 is frequently non-linear because the value of ϵ (or more frequently μ for magnetic fields) is dependent on the field intensity.

Non-Linear Materials

A Newton-Raphson method can be used to solve this type of non-linear equation. Given an initial solution Φ_n a new solution Φ_{n+1} is found by solving the linearized Jacobean system¹:

$$\Phi_{n+1} = \Phi_n - J_n^{-1} R_n \quad (6.12)$$

where the residual R is given by

$$R_n = K_n \Phi_n - S_n \quad (6.13)$$

and the Jacobean J by

$$J_n = \frac{\partial}{\partial \Phi_n} (K_n \Phi_n - S_n) \quad (6.14)$$

This method converges providing the initial approximation used to start the iteration is not too far from the real solution. As it approaches the solution its convergence becomes quadratic. In the context of non-linear finite element solutions to the electromagnetic field equations the reliability of the Newton-Raphson method is strongly linked to the smoothness of the equation used to relate the permeability or permittivity to the field.

To aid convergence, when the material property curve is not smooth or when the initial solution is far from the final solution, a relaxation factor is used in equation 6.12, which becomes:

$$\Phi_{n+1} = \Phi_n - \alpha J_n^{-1} R_n \quad (6.15)$$

where α is chosen, starting with 1 and multiplying by 2 (if the change in $|R|$ is too small) or dividing by 2 (if the norm of the residual $|R_{n+1}|$ would be greater than $|R_n|$) to find the value which minimises the norm of the residual $|R_{n+1}|$ at the start of the next iteration.

1. The subscript 'n' indicates the iteration number

Finite Element Applications

The experience of the user has been a vital ingredient in the successful application of finite elements to predictive engineering design. Large finite element systems for mechanical design have a mystique associated with them, partly from their origins as stand alone programs with a text file user input interface and a box of paper as the output display, and partly as a result of the jargon that is used to describe the element types and procedures available within the programs. It is now impossible to consider the use of finite element analysis programs without interactive pre processors for data input, although these bring their own problems if they do not interface well with the analysis programs and thus increase the amount of knowledge needed to perform a calculation.

In electromagnetic field calculations special finite elements are not needed to solve the equivalent of shell and plate geometries that are so common in mechanical design. However, electromagnetic fields must usually be computed to much higher accuracy than is needed in other disciplines, the geometry is frequently complicated with a wide range of dimensions and the actual result needed by the user is often derived from the field solution by integration or differentiation. In whatever form the results are required, the basic limitation of finite element solutions is that the accuracy of the solution is related to the size of the discrete elements. Recent research has resulted in the development of techniques that can be used to determine the error in a finite element solution but this is strictly only correct for problems with linear materials. The adoption of these techniques in finite element programs will improve the reliability of results, but they do not help to check that the finite element model and the physical problem are equivalent.

Codes of Practice

Users of finite element programs must prove that the model is consistent with the physical problem. With electromagnetic fields it is often possible to perform simple calculations that give *orders of magnitude* answers as an essential part of the analysis. Until the accuracy of the model has been established it is irrelevant to consider the discretisation errors. A code of practice should be established that is followed whenever a new analysis is begun:

1. If it is appropriate solve a simplified two dimensional model of the system, apply tests 2,3 and 4 (below) to that solution and then use the two dimensional solution as an initial check of the full three dimensional solution.
2. Once the three dimensional model has been defined, solve the simplest possible problem i.e. using linear materials either with unity or large relative permeability or permittivity.

3. Check that the solution has the symmetry that is expected. For example, examine the fields on the boundaries of the problem to see if they are as expected.
4. Check that the solution agrees with simple line integral predictions or images if infinite permeability approximations are applicable.

Only when a degree of confidence has been established in the model is it worth beginning to consider the errors produced by the discrete finite element approximation. In many ways these errors are more straightforward to evaluate than the accuracy of the model.

Solution Errors

The local error at a point within a finite element model is strongly linked to the size of the elements surrounding the point and weakly linked to the average element size over the whole space, although this second source of error becomes more important and less easily estimated in non-linear solutions. The relationship between the local error in the solution and the surrounding elements' size is given by:

For linear shape functions

$$E(\Phi) = O(h^2) \quad (6.16)$$

and for quadratic shape functions

$$E(\Phi) = O(h^3) \quad (6.17)$$

where E is the error, O means 'of the order' and h is the linear dimension of the elements. This simple analysis is only true for square elements, but it is reasonable to assume the worst case and use the largest dimension for h . Unfortunately, these formulae only give the order of the error, the actual error is dependent on the solution, or more precisely the geometry of the model in the vicinity of the point. As an example consider a point close to the corner of a magnetised steel cube, the field will be weakly singular at the corner. Given the same size discretisation over the whole space, the errors will be far larger close to the edges and corners of the cube. This is because the low order polynomials used in the finite elements are not good at approximating the singularity. Calculating the magnetic field from the potential solution generally results in larger errors in the field than there were in the potential.

Differentiation of the finite element shape functions to determine the field gives an error in the field that is worse by $O(h^{-1})$. In the case of linear shape functions this results in an error in the field $O(h)$. In the analysis modules special facilities

have been included in order to reduce the errors in the fields that are computed from potential solutions. Two methods are available that increase the field precision; the best method depends on the problem being solved (see sections on accuracy in the following chapters). Nodal weighted averaging improves the field accuracy to $O(h^2)$. The volume integration technique does not improve the order of the error, but it enables the variation of the field to be calculated very accurately remote from magnetic, dielectric or conducting regions.

The programs use error estimation techniques to produce local and global errors in the fields derived from potential solutions (See [“Solution Accuracy” on page 3-9.](#)). These displays show where the finite element discretisation needs refinement as well as showing the error. However, even with these features, it is important that the user of a finite element program carries out a number of analyses to examine the effect of element size on the solution. Using the ideas introduced above it is clear that the best approach is to solve the same model with two levels of finite element discretisation or with the same discretisation but using linear elements in one case and quadratic elements in the other. Taking as an example the use of two levels of discretisation, such that the element dimensions are halved in the second case, the case with the larger number of elements will have solution errors that are 4 times smaller (the errors in the fields evaluated by differentiation of the shape functions will be halved). Examination of the changes between the two solutions will give a good estimate of the discretisation errors, but not, as pointed out in the previous section, any indication of the accuracy of the model.

This approach is very good for two dimensional discretisations, but in three dimensions the 8 fold increase in the number of nodes quickly becomes prohibitive for all but the simplest geometries. When increasing the overall discretisation becomes too expensive it is necessary to carry out more trial analyses, in each case choosing particular regions of increased discretisation to determine the sensitivity of the solution to the change in element size. An experienced user will have learnt how to minimise the number of trials as a result of carrying out this type of experiment on a number of different geometries. Unfortunately a little knowledge may be a dangerous thing! Even experts cannot rely completely on their past experience. Results must always be critically examined on the assumption that they are incorrect. The adoption of error analysis techniques in finite element programs does not reduce the user’s responsibility for the quality of the results.

TOSCA, Static Field Analysis

Introduction

This section describes the electromagnetic field analysis program TOSCA. The program is based on a program of the same name which was developed at the Rutherford Appleton Laboratory in England. It represents the results of many years research, development and application experience. TOSCA has been further extended and is now one of the analysis programs of the OPERA-3d Vector Fields Electromagnetic Analysis Environment.

TOSCA can be used to compute current flow, magnetostatic or electrostatic fields, including the effects of non-linear media in three dimensions. The program incorporates state of the art algorithms for the calculation of electromagnetic fields, advanced finite element and non-linear equation numerical analysis procedures. In the following sections, the algorithms used in TOSCA are described so that users are able to relate the finite element model to their application problem.

The TOSCA Algorithm

TOTAL and REDUCED Magnetic Scalar Potentials

Three dimensional stationary electromagnetic fields can be represented as the sum of a solenoidal field and a rotational field. In electrostatic fields there is never a rotational component, the field can therefore always be defined using the electrostatic potential (V). The electric field intensity (\mathbf{E}) is given by:

$$\mathbf{E} = -\nabla V \quad (6.18)$$

The divergence of the electric flux density (\mathbf{D}) is related to the charge density (ρ):

$$\nabla \cdot \mathbf{D} = \rho \quad (6.19)$$

Combining equations 6.18 and 6.19 and introducing the dielectric permittivity tensor (ϵ) gives the usual Poisson's equation description of the electrostatic potential:

$$\nabla \cdot \epsilon \nabla V = -\rho \quad (6.20)$$

where $\mathbf{D} = \epsilon \mathbf{E}$.

A similar equation arises for current flow problems,

$$\nabla \cdot \sigma \nabla V = 0 \quad (6.21)$$

where σ is the conductivity, and $\mathbf{J} = \sigma \mathbf{E}$.

Stationary magnetic fields on the other hand in general consist of both solenoidal and rotational components. The field produced by electric currents has a rotational component inside the volumes where the currents flow. In the exterior space the field is solenoidal but the scalar potential is multi-valued. The field produced by magnetised volumes is solenoidal. It is convenient to use a splitting of the total field into two parts in order to obtain a description of the field in terms of a simple scalar potential.

The total field intensity \mathbf{H} is defined using the reduced field intensity (\mathbf{H}_m) and the conductor field intensity (\mathbf{H}_s):

$$\mathbf{H} = \mathbf{H}_m + \mathbf{H}_s \quad (6.22)$$

The reduced field intensity can now be represented using the reduced scalar potential (ϕ):

$$\mathbf{H}_m = -\nabla \phi \quad (6.23)$$

and in the case of stationary magnetic fields where the exciting currents are prescribed, the conductor field intensity may always be directly evaluated by integration:

$$\mathbf{H}_s = \int_{\Omega_J} \frac{\mathbf{J} \times \mathbf{R}}{|\mathbf{R}|^3} d\Omega_J \quad (6.24)$$

The divergence of the flux density is always zero. Introducing the permeability tensor (μ) and combining equations 6.22 to 6.24 gives the partial differential equation for the reduced scalar potential (ϕ):

$$\nabla \cdot \mu \nabla \phi - \nabla \cdot \mu \left(\int_{\Omega_J} \frac{\mathbf{J} \times \mathbf{R}}{|\mathbf{R}|^3} d\Omega_J \right) = 0 \quad (6.25)$$

This equation, like the Poisson's equation for electrostatic fields, can easily be solved using the finite element method. However the reduced potential formulation for magnetic fields is not acceptable. Large errors are found in the total fields computed by this method.

The errors arise because the space variation of \mathbf{H}_m and \mathbf{H}_s will be quite different if one is represented using derivatives of the finite element shape functions (a low

order polynomial in x) and the other using direct evaluation of the integral in equation 6.24. This effect combined with the possibility that \mathbf{H}_m and \mathbf{H}_s will strongly cancel in some volumes of the space leads to magnification of the errors in the approximate solution for \mathbf{H}_m .

The cancellation problem is particularly critical in the interior of non-linear magnetic materials, where the magnified errors destroy the accuracy of the Jacobean matrix used for Newton iterations. It is also very undesirable when magnetic shielding is being designed since the largest errors will occur in volumes where the shield is most effective.

The above difficulty can be completely avoided when currents are not flowing in the magnetic materials. Exterior to the volumes where currents flow the total field can be represented using the total magnetic scalar potential (ψ):

$$\mathbf{H} = -\nabla\psi \quad (6.26)$$

where the total magnetic scalar potential satisfies:

$$\nabla \cdot \mu \nabla \psi = 0 \quad (6.27)$$

By combining the two representations (the total and reduced scalar potentials) cancellation difficulties can be completely avoided. The minimal combination consists of using the reduced potential only inside volumes where currents flow and the total potential everywhere else. This has practical limitations in that the reduced potential volume may have a complicated shape and cutting surfaces would need to be specified in the total potential space to maintain a single valued potential. For these reasons simple singly connected spaces are generally used for the reduced potential volumes. Facilities exist to perform the cuts automatically within the software - see [“Multiply Connected Regions” on page 7-15](#) for more information.

In exceptional circumstances, where it is impossible to create singly connected total scalar potential volumes, it is possible to use reduced scalar potential within magnetic materials. Because of the cancellation problems described above, such reduced scalar potential volumes should be as small as possible and away from regions where accuracy is critical. Reduced scalar potential must be used in magnetic material carrying source currents.

On the interface between the total and reduced potential spaces the two potentials can be exactly linked together by applying the conditions of normal \mathbf{B} and tangential \mathbf{H} continuity. This involves evaluation of the normal field produced by the conductors and the scalar potential that could be used to represent the conductor field on the interface surface. Practically, this also makes the method more attractive than the straightforward reduced potential formulation. Evaluating the fields

from conductors although well defined, is expensive. The reduced potential formulation requires that the conductor fields be evaluated throughout the non-linear magnetic material volumes. The total potential on the other hand only requires the conductor field to be evaluated on the interface surface between the reduced and total potential volumes for the solution of the equations. However, the conductor field is needed throughout the reduced potential volumes to obtain the total field during post processing. It is also needed within volumes defined as reduced potential within magnetic regions.

Boundary Conditions

Boundary conditions are used in two ways. Firstly they can provide a way of reducing the size of the finite element representation of symmetrical problems. Secondly they are used to approximate the magnetic field at large distances from the problem (far-field boundaries).

General

Problem symmetry and the symmetry of the fields are implied by the potential boundary conditions applied to the finite element model. The simplest types of boundary condition are shown in the following table (where \mathbf{n} is the normal unit

Table 6.1 Boundary Conditions

Magnetic Fields	Field Symmetry	Scalar Potential
TANGENTIAL MAGNETIC	$\mathbf{H} \cdot \mathbf{n} = 0$	$\frac{\partial \phi}{\partial n} = 0$
NORMAL MAGNETIC	$\mathbf{H} \times \mathbf{n} = 0$	$\phi = \text{constant}$
Electric Fields or Current Flow	Field Symmetry	Scalar Potential
TANGENTIAL ELECTRIC	$\mathbf{E} \cdot \mathbf{n} = 0$	$\frac{\partial V}{\partial n} = 0$
NORMAL ELECTRIC	$\mathbf{E} \times \mathbf{n} = 0$	$V = \text{constant}$

vector to the surface being considered). Note that in the table, ϕ refers to either the reduced or total scalar potential. Boundary conditions on the reduced scalar potential only affect the reduced field intensity.

In electrostatic fields electrode surfaces will obviously have assigned potential boundary conditions ($V = \text{value}$). In the analogous current flow problem (for which TOSCA may also be used), it may also be necessary to assign non-zero derivative boundary conditions ($\partial V / \partial n = \text{value}$) to define impressed currents. Mixed bound-

ary conditions defined by $V + \alpha \frac{dV}{dn} = \beta$ are also available to model polarisation potentials produced by corrosion of metal surfaces. Except for these cases, the other boundary conditions shown in the previous table above should only be applied to the exterior surfaces of the finite element model. The default condition that will always be applied if no boundary condition is specified on an exterior surface is:

Table 6.2 Default Boundary Conditions

Magnetic Fields	Field Symmetry	Scalar Potential
TANGENTIAL MAGNETIC	$\mathbf{H} \cdot \mathbf{n} = 0$	$\frac{\partial \phi}{\partial n} = 0$
Electric Fields or Current Flow	Field Symmetry	Scalar Potential
TANGENTIAL ELECTRIC	$\mathbf{E} \cdot \mathbf{n} = 0$	$\frac{\partial V}{\partial n} = 0$

Nodal Field Extraction

The default boundary condition is only weakly satisfied even when it is assigned to a surface. ‘Weakly satisfied’ means that it is applied as an integral over the surface patch for each shape function sub-domain. Examination of the field solution close to a surface with the default boundary condition will reveal that the normal component is not zero. The magnitude of the normal component reflects the local accuracy of the solution unless it is the result of a model error. Such an error may arise in a reduced potential region if the conductor’s field does not have the same symmetry as the finite element model. On the surface itself however, the normal component is forced to be precisely zero by the post processor.

The fields computed by taking derivatives of the finite element shape functions will be discontinuous, but in each element the potential boundary condition will be exactly specified. The field smoothing processes in TOSCA (nodal averaging, see the notes on accurate field computation in section “**Accurate Fields**” on [page 6-16](#)) must take this into account, and hence the true boundary condition is forced at these surfaces to ensure accuracy is maintained as far as possible.

Uniqueness

A potential boundary condition must be specified on at least one surface of a scalar potential problem. This gauges the scalar potential and without it the solution will not be unique. Any constant value could be added over the whole space, although the implementation in TOSCA constrains zero to be the only valid assigned potential boundary condition within reduced potential volumes. In

TOSCA this condition is satisfied by one surface in the problem having an assigned potential.

If no surface is assigned a value for the potential, one point is arbitrarily chosen by the analysis to have a zero potential, enforcing a unique solution.

Open Boundaries

Electromagnetic fields are frequently not contained within a finite volume. In practice, at long distances from the device producing the field, the distribution will be modified by the local environment, but this will not effect the field close to the device. Except where the interaction with the far environment is of interest, the field from an isolated device is usually required. A simple finite element mesh obviously has a finite extent and applying either potential or derivative boundary conditions on the open boundaries will perturb the true infinite domain solution.

There are techniques that accurately model the infinite domain. On a convex outer surface a series of rings of elements with increasing size may be recursively generated automatically (ballooning) or a boundary integral solution for the exterior domain may be coupled to the interior finite element solution. These techniques are relatively expensive in three dimensions and approximate methods may be used instead. The approximate methods include matching a convex outer surface to finite elements that extend from the surface to infinity and which have appropriate decay functions, and the standard approach of extending the finite element mesh to a distance where the field truncation has no effect on the regions of interest.

It is recommended that the mesh be extended to a reasonable distance with either potential or derivative boundary conditions applied to the outer surface, so that the truncation has an insignificant effect on the region of interest. The effect of truncation can be estimated by observing the tangential component of the field on an open boundary with a derivative boundary condition. Half the field observed on such a boundary is being reflected back from the exterior. Combining this with knowledge of the probable decay in the exterior space will give an order of magnitude for the effect of the truncation on the regions of interest. In a particularly sensitive application a further test should be applied. Two problems should be solved, one using potential boundary conditions on the open boundary, the other using derivative conditions. These problems represent the model in an infinite array of similar problems with either the opposite or the same sign of field in alternate images. In general the two solutions will bound the correct answer. It is often found that the derivative condition is closer to the real solution. It would in fact be exact if the open boundary surface was a constant flux surface. Use of this simple approach gives an estimate for the effect of the far field boundary truncation and this is why it is recommended.

Total and Reduced Potential at the Open Boundary

It is worthwhile considering the significance of the potential type interior to the open boundary surface. With a total potential interior to the open boundary surface both the conductor and iron fields will be reflected at the surface. On the other hand, if a reduced potential is interior to the open surface only the iron fields will be reflected at the surface. This effect may be used to minimise the perturbation caused by the truncation.

For example, if the problem consists of a small volume of iron, interior to a large conductor system, the reduced potential at the open surface will clearly give much smaller errors produced by the approximate open (far field) boundary condition. The relatively small field from the iron will be the only component reflected in the false open boundary.

An alternative example is one where the conductors are contained inside an iron shield. In this case the total field at the open boundary will be much smaller than the conductors' field (if the shield is effective), a total potential next to the open boundary surface will therefore give much less perturbation in the regions of interest.

Periodicity Conditions

The potential and derivative boundary conditions discussed in the previous sections are the most common in nearly all applications. There is one other class of boundary condition that frequently occurs in electrical machine design. When the problem has rotational symmetry about an axis or displacement symmetry, without any reflection symmetry in the same symmetry group, it is not possible to identify surfaces where the field is normal or tangential. The complete model could be defined but this would be unnecessarily expensive. In problems with this type of symmetry, pairs of matching surfaces can be identified where the potential values have the same sign and magnitude (positive) or the reverse sign but equal magnitude (negative). In terms of fields, for rotational electric machine problems, this translates to the radial and azimuthal components of field on the boundaries being identical (positive) or equal but opposite (negative).

The surface pairs must possess an exact, linear, point matching mapping - complete congruence is essential. This congruence of the surface pairs applies not only to the field symmetry but also to the finite element discretisation used on the surfaces, if periodicity conditions are to be used with TOSCA.

The surface pairs must be identified before the finite element model is defined so that the two surfaces can be defined with congruent discretisations. In TOSCA the surfaces with periodic boundary conditions are marked as having symmetry boundary conditions, and a geometrical symmetry group operation is defined that makes the surface pairs exactly overlap. The symmetry group operations are rotations, displacements or a combination of rotation followed by displacement. The symmetry group operation for a particular surface pair should transform the nodes

from one of the surfaces so that they become coincident with their matching nodes on the other surface. The sign of the periodicity condition (positive or negative) is defined with the symmetry group operation.

Although the above discussion concentrated on pairs of surfaces, the technique can be applied to any number of surfaces. A set of symmetry group transformations can be defined, one of which may bring two symmetry surfaces into coincidence, another may make a third surface coincide with one of the first pair. In general this higher order periodicity is only found under displacement group operations.

Since the periodicity condition matches all points on one surface to those on another, it is essential that the distribution of points on each surface is identical. This can only be ensured if the faces with periodic boundary conditions have regular surface meshes.

Accurate Fields

The total and reduced potential formulation used in TOSCA allows a magnetostatic problem to be defined with many possible combinations of potentials. The only restriction is that the total potential should be used inside magnetic volumes and the reduced potential inside volumes where currents are flowing. Apart from these restrictions, the distribution of potential types may be chosen to give optimum accuracy for particular applications. One aspect of this has been mentioned in section [“Total and Reduced Potential at the Open Boundary” on page 6-15](#) and [“Boundary Conditions” on page 6-12](#). The following table identifies particular classes of problem, and shows the recommended potential type in a particular volume of space.

In the table: ‘Nodal’ refers to the nodally averaged field output option; ‘Integral’ to the option where the field from the magnetic volumes are computed by evaluating the magnetisation from the potential solution and then computing the field by volume integration of the magnetisation.

Table 6.3 Potential Types

Problem Description	Potential Type	Field Recovery	
		from mesh	from conductors
Homogeneous Field Conductor field » Iron field Close to the iron Far from the iron	Reduced Any	Nodal Integral	Integral Integral
Homogeneous Field Conductor Field \approx Iron Field Homogeneous conductor field Inhomogeneous conductor field or if total cannot be used	Any Total Reduced	Nodal Nodal Nodal	Integral Integral Integral
Homogeneous Field Conductor Field \approx -Iron Field Homogeneous conductor field Inhomogeneous conductor field or if total cannot be used	Total Total Reduced	Nodal Nodal Nodal	Any Any Integral
Homogeneous Field Conductor Field « Iron Field	Any	Nodal	Integral
Peak Field on Conductor	Reduced	Nodal	Integral
Inhomogeneous Field Conductor Field » Iron field Conductor Field « Iron field As above but far from iron Conductor Field \approx Iron field Conductor Field \approx -Iron field	Reduced Any Any Any Total	Nodal Nodal Integral Nodal Nodal	Integral Integral Integral Integral Integral

The table does not cover all eventualities and it may help to explain why the combinations are recommended. First, the use of total potential in volumes where the conductor and iron fields cancel is the basis on which the TOSCA algorithm was designed. This takes precedence over all other considerations. Finite elements are good at representing uniform fields and by implication, not so good for varying fields. If the combined conductor and iron fields produce a uniform resultant field, then it is best to use the total potential in such volumes. However, if the conductor field is much greater than the iron field, it is best to use a reduced potential in this volume because the conductor fields are essentially exact.

When TOSCA results are displayed using the OPERA-3d post processor, the effects of the various types of field evaluation can be tested. The error caused by using interpolated conductor fields can be immediately compared against the

more accurate but expensive conductor integration procedure. It is recommended that when peak fields on conductors or body forces on conductors are required, then conductor fields should be calculated by integration.

The fields computed by magnetisation integration are a very specialised option. The primary role of this option is in applications such as shielded superconducting NMR systems. Field accuracy of the order of parts in 10^6 are required at some distance from the shield and where the shield contributes less than 10% of the total field. Reduced potential could be used in these volumes, providing the conductor fields are uniform. This can achieve an accuracy of 3 in 10^5 . Comparison with analytical solutions has showed that the magnetisation integration can improve this to 3 in 10^6 . However this procedure is very computationally expensive.

Magnetic Shielding

Many of the recommendations in the sections on accuracy apply to the calculation of electromagnetic fields in applications where shielding is the primary objective. The most important recommendations for shielding are that in the volumes where the shield is effective it is essential that the total potential option is used and fields are evaluated either from the shape function derivatives, or better still from the nodally averaged field.

When the shield is screening the outside world from the field produced by a device the above recommendation means that a total potential will be used in the exterior space. This is also important from the point of the approximate far field boundary condition that will be imposed at the boundary of the mesh. The total potential gives minimal perturbation of the interior solution.

Thin magnetic screens are often used at large distances from the device to be isolated. Elements with large aspect ratios can be used in such screens and will not adversely affect the solution accuracy. However, if a detailed evaluation of the field at a corner or next to a small gap is required, then the elements' dimension must be small compared to the feature being modelled. The restriction is perfectly reasonable, in any direction where the field is changing rapidly the element discretisation must be capable of modelling the changes.

ELEKTRA, Time Varying Analysis

Introduction

This section describes the electromagnetic field analysis program ELEKTRA, which is one of the analysis programs of the OPERA-3d Vector Fields Electromagnetic Analysis Environment.

ELEKTRA can be used to compute electromagnetic fields including the effects of eddy currents, in three dimensions. The program incorporates state of the art algorithms for the calculation of electromagnetic fields and advanced finite element numerical analysis procedures. In the following sections the algorithm used in ELEKTRA is described so that users are able to relate the finite element model to their application problem.

The ELEKTRA Algorithm

ELEKTRA can use a combination of vector and scalar magnetic potentials to model time varying electromagnetic fields. Vector potentials have to be used in conducting media; scalar potentials can be used in the rest of space (with certain restrictions that depend on the geometry).

In general, scalar potentials are preferred since they involve only a single unknown at each node in the problem, compared to four when using the vector potential. This means a large saving in memory requirements, and also indirectly in computation time.

Fields in Source Free, Non-Conducting Media

In a region of free space that does not include source currents, it is generally found to be most efficient to replace the magnetic field by the gradient of a magnetic scalar potential, ψ , so that:

$$\mathbf{H} = -\nabla\psi \quad (6.28)$$

In a similar way to the TOSCA algorithm outlined for electrostatic problems in section “TOTAL and REDUCED Magnetic Scalar Potentials” on page 6-9, this leads to a governing equation:

$$\nabla \cdot \mu \nabla \psi = 0 \quad (6.29)$$

where $\mathbf{B} = \mu\mathbf{H}$.

Fields in Non-Conducting Media With Sources

Fields in non-conducting media with sources in general consist of both solenoidal and rotational components. The field produced by known electric (source) currents has a rotational component inside the volumes where the currents flow. In the rest of free space the field is solenoidal but the scalar potential is multi-valued. The field produced by magnetized volumes is solenoidal.

It is convenient to use a splitting of the total field into two parts in order to obtain a description of the field in terms of a simple scalar potential in non-conducting media. The total field intensity \mathbf{H} is defined using the reduced field intensity ($\mathbf{H}_m + \mathbf{H}_e$) and the conductor field intensity (\mathbf{H}_s):

$$\mathbf{H} = \mathbf{H}_m + \mathbf{H}_e + \mathbf{H}_s \quad (6.30)$$

The reduced field intensity has two contributions, one from magnetisation fields in iron (\mathbf{H}_m), and the other due to fields from induced eddy currents in conductors (\mathbf{H}_e). The sum of these two is the unknown quantity, and can now be represented using the reduced scalar potential (ϕ):

$$\mathbf{H}_m + \mathbf{H}_e = -\nabla\phi \quad (6.31)$$

In the case of stationary magnetic fields where the exciting currents are prescribed, the conductor field intensity may always be directly evaluated by integration:

$$\mathbf{H}_s = \int_{\Omega_J} \frac{\mathbf{J} \times \mathbf{R}}{|\mathbf{R}|^3} d\Omega_J \quad (6.32)$$

For further details of the use of total and reduced scalar potentials in non-conducting media, look at section [“TOTAL and REDUCED Magnetic Scalar Potentials” on page 6-9](#).

Fields in Conducting Media

In time varying fields the currents that are induced in conducting volumes are part of the unknowns in the system. Their fields cannot therefore be evaluated by simply performing an integration. Inside the conducting volumes the field representation must include a rotational component. The most elegant approach is to combine the efficient total and reduced scalar potential method for non-conducting volumes with an algorithm that uses a vector potential (\mathbf{A}) in the conducting volumes.

In a low frequency time varying magnetic field when the dimensions of the objects in the space are small compared to the wavelengths of the fields, the magnetic and electric fields are related by the low frequency limit of Maxwell's equations:

$$\nabla \times \mathbf{H} = \mathbf{J} \quad (6.33)$$

$$\nabla \times \mathbf{E} = -\frac{\partial \mathbf{B}}{\partial t} \quad (6.34)$$

where

$$\mathbf{J} = \sigma(\mathbf{E} + \mathbf{u} \times \mathbf{B}) \quad (6.35)$$

and σ is the electrical conductivity and \mathbf{u} the velocity. At this point we assume that the velocity is zero. The affect of non-zero velocity will be discussed in section “[The Motional Equations](#)” on page 6-25 below.

The flux density \mathbf{B} can be represented using a vector potential \mathbf{A} where:

$$\mathbf{B} = \nabla \times \mathbf{A} \quad (6.36)$$

Using this vector potential and combining equations 6.33 to 6.36 gives the following equation for \mathbf{A} :

$$\nabla \times \frac{1}{\mu} \nabla \times \mathbf{A} = -\sigma \frac{\partial \mathbf{A}}{\partial t} - \sigma \nabla V \quad (6.37)$$

The electric scalar potential (V) emerges because of the non-uniqueness of the potential which arises during integration of equation 6.33.

To ensure a unique solution therefore, it is also necessary to ‘gauge’ equation 6.37. The gauge used in ELEKTRA is the Coulomb gauge

$$\nabla \cdot \mathbf{A} = 0 \quad (6.38)$$

and after including it into equation 6.37 the new governing equation is

$$\nabla \times \frac{1}{\mu} \nabla \times \mathbf{A} - \nabla \frac{1}{\mu} \nabla \cdot \mathbf{A} + \sigma \left(\frac{\partial \mathbf{A}}{\partial t} + \nabla V \right) = 0 \quad (6.39)$$

It is also necessary to solve the secondary equation

$$\nabla \cdot \sigma \nabla V + \nabla \cdot \sigma \frac{\partial \mathbf{A}}{\partial t} = 0 \quad (6.40)$$

This vector potential description of fields inside conducting volumes can be directly combined with the total and reduced potential description in non-conducting spaces. The normal flux and tangential field intensity interface conditions are used to exactly couple the two field descriptions.

Time Variation in ELEKTRA

There are three ELEKTRA analysis modules, each having a different form of time variation.

- ELEKTRA-SS calculates steady-state ac currents (the time harmonic form) where all fields and potentials are oscillating at the same frequency.
- ELEKTRA-TR calculates transient eddy currents induced by the fields of driving currents which change in time in a predetermined way.
- ELEKTRA-VL calculates eddy currents induced by motion which does not change the geometry of the problem.

The Time Harmonic Equations

Under steady state alternating current excitation a complex potential substitution can be used. This carries the assumption that the materials have linear characteristics and that therefore the waveforms of the current and field are precisely the same. The substitutions are:

$$\begin{aligned}\mathbf{A}(t) &= \mathbf{A}_c e^{i\omega t} \\ \Phi(t) &= \Phi_c e^{i\omega t} \\ V(t) &= V_c e^{i\omega t}\end{aligned}\tag{6.41}$$

where ω is the angular frequency of the alternating current excitation. With this assumption equation 6.39 becomes

$$\nabla \times \frac{1}{\mu} \nabla \times \mathbf{A}_c - \nabla \frac{1}{\mu} \nabla \cdot \mathbf{A}_c + i\omega \sigma \mathbf{A}_c + \sigma \nabla V_c = 0\tag{6.42}$$

and equation 6.40 becomes

$$\nabla \cdot \sigma \nabla V_c + i\omega \sigma \nabla \cdot \mathbf{A}_c = 0\tag{6.43}$$

The form of the equations in magnetic scalar potential is not affected by this substitution.

Quasi Non-Linear Solutions

Although it has been assumed that the materials are linear, it is still possible to model non-linear materials using a quasi non-linear model.

The assumption is that \mathbf{B} and \mathbf{H} are both in phase, so that 6.41 above is still valid. The magnitude of the field is used to determine the value of the permeability (based on the usual BH relationship), which is then used in equation 6.42. This is a non-linear process, and a simple update scheme is used— having solved equation

6.42, the element permeabilities are updated based on the BH curve, and the equation re-solved, until a converged solution is obtained.

The Transient Equations

The Transient Analysis Program (ELEKTRA-TR) solves eddy current problems where the driving currents are changing in time in a predetermined way, but not necessarily with a time harmonic behaviour.

If the Galerkin procedure is applied to the terms in \mathbf{A} and $\frac{\partial \mathbf{A}}{\partial t}$ in equation 6.16 both yield matrices, referred to as \mathbf{R} and \mathbf{S} . This leads to a matrix equation

$$\mathbf{R}\mathbf{A} + \mathbf{S}\frac{\partial \mathbf{A}}{\partial t} + \mathbf{B} = 0 \quad (6.44)$$

where \mathbf{A} is now a vector of unknown potentials and \mathbf{B} is a vector of driving terms. The solution of equation 6.44 is also based on the Galerkin procedure. \mathbf{A} and \mathbf{B} are discretised in time using a first order function of time:

$$\mathbf{A}(t) = (1 - \tau)a_n + \tau a_{n+1} \quad (6.45)$$

$$\mathbf{B}(t) = (1 - \tau)b_n + \tau b_{n+1} \quad (6.46)$$

where

$$\tau = \frac{t - t_n}{t_{n+1} - t_n} \quad (6.47)$$

and a_n and b_n are values of \mathbf{A} and \mathbf{B} at time t_n . Using τ as the weight in a Galerkin weighted residual solution of 6.44 leads to a recurrence relationship between a_{n+1} and a_n :

$$\left(\mathbf{R}(1 - \theta) - \frac{\mathbf{S}}{\Delta t}\right)a_n + \left(\mathbf{R}\theta + \frac{\mathbf{S}}{\Delta t}\right)a_{n+1} + b_n(1 - \theta) + b_{n+1}\theta = 0 \quad (6.48)$$

where $\theta = \frac{2}{3}$ (and $\theta = 1$ for non-linear problems) and the time step

$$\Delta t = t_{n+1} - t_n.$$

There are three options available to select the time step:

- the time step is a constant value for the duration of the analysis. However, this can be changed by restarting the solution, using a different value of time step.

- the time step is a constant value for the duration of the analysis, but is further subdivided into four equal sub time steps and a fourth order Runge-Kutta method is used to improve the time interpolation
- the length of the time step is adjusted adaptively by ELEKTRA-TR using a fourth order Runge-Kutta method to achieve a user-defined accuracy of solution at each step

The type of time stepping algorithm is selected by the user in the pre processor and the Modeller as an option presented during creation of the analysis data for an ELEKTRA-TR simulation.

Transient Driving Functions

The 'driving' field is provided by source currents, J_s , or non-zero potential boundary conditions. The available drive functions within ELEKTRA-TR are as follows:

Table 6.4 Transient Drive Functions

Name	Function
DC	$F=1$ for all time
STEP	$t < 0: F=0$ $t \geq 0: F=1$
RAMP	$t < 0: F=0$ $0 \leq t \leq t_c: F = \frac{t}{t_c}$ $t > t_c: F=1.$
SINE	$t < 0: F=F(0)$ $t \geq 0: F=\sin 2\pi ft + \phi.$
COSINE	$t < 0: F=F(0)$ $t \geq 0: F=\cos 2\pi ft + \phi.$
PEAK	$t < 0: F=0$ $t \geq 0: F = a \exp(-t^2/b)$ a and b are chosen such that $F=1$ at $t=t_c.$
RISE	$t < 0: F=0$ $t \geq 0: F = 1 - \exp(-t/t_c)$
TABLE (switch on)	$t < 0: F=0$ $t \geq 0: F=\text{cubic splines.}$
TOFF (switch off)	$t < 0: F=F(0)$ where $F(0)$ is the value of the function in the table file at time $t=0.$ $t \geq 0: F=\text{cubic splines.}$

The driving function can be applied to all the source conductors and boundary conditions, or the drive label can be used to specify different driving functions for different conductors and boundary conditions. A phase angle can also be used in the sine and cosine functions.

The transient time table option allows the user to define driving functions other than those programmed into the analysis code. The tables consist of files containing pairs of numbers in free format, one pair per line. The first number on each line specifies the time; the second gives the function value. A total of 4096 pairs of points are allowed, distributed arbitrarily between the time table files. To check the shape of the drive function, the **GRAPH** command can be used to display the file contents.

For the time table **TABLE** “Switch On” option, the values of time should start at zero and increase through the file. Discontinuities in function value or first derivative can be forced by specifying two entries for the same value of time. For the time table **TOFF** “Switch Off” option, the value of the function is set to the value of the drive function in the time table at $t=0$ for all time up to time zero.

Other functions (**DC**, **COSINE**, and **SINE**) will also have function values that are non-zero at time zero (depending on the phase angle for sine and cosine drives).

Beyond the last value of time in the table, the function continues with the last value computed.

The Motional Equations

The analysis program ELEKTRA-VL solves the static field problem, including the effect of motionally induced eddy currents. The conducting parts of the problem can be assigned a velocity. This can be a linear velocity, specified by speed and direction or a rotational velocity specified by the angular velocity around the global z-axis. In either case, in each element there is a velocity vector, \mathbf{u} . It is always assumed that the driving coils are stationary and the eddy current conductors in motion. The formulation of ELEKTRA-VL assumes that, at every instant in time, the geometry of the model is identical. This means that the cross-section of the moving conductor orthogonal to the direction of motion does not change. In other words, for linear motion the moving conductor is “infinite” and for rotational motion the RZ cross-section is invariant. Examples of these might be a pipeline inspection vehicle and an eddy current disk brake respectively.

From equations 6.35 and 6.36, the current is related to the potentials \mathbf{A} and V by

$$\mathbf{J} = \sigma(\mathbf{u} \times \nabla \times \mathbf{A}) - \sigma \nabla V \quad (6.49)$$

where \mathbf{u} is the velocity. The equation that is solved (based on 6.39) is

$$\nabla \times \frac{1}{\mu} \nabla \times \mathbf{A} - \nabla \frac{1}{\mu} \nabla \cdot \mathbf{A} = \sigma(\mathbf{u} \times \nabla \times \mathbf{A}) - \sigma \nabla V \quad (6.50)$$

and equation 6.40 becomes

$$\nabla \cdot \sigma \nabla V - \sigma \nabla \cdot (\mathbf{u} \times \nabla \times \mathbf{A}) = 0 \quad (6.51)$$

Non-Linear Materials

All the analysis modules in ELEKTRA now include non-linear material properties, which can also be anisotropic. In the ELEKTRA-VL and ELEKTRA-TR analysis modules, permanent magnets can be included, and the behaviour of these materials is also defined by BH curves that extend into the 3rd quadrant.

In each case, a direct iteration method is used (rather than Newton Raphson used in TOSCA). This implies that in general more iterations will be required to converge to a non-linear solution.

Boundary Conditions

Boundary conditions are used in two ways. Firstly they can provide a way of reducing the size of the finite element representation of symmetrical problems. Secondly they are used to approximate the magnetic field at large distances from the problem (far-field boundaries).

General

Problem symmetry and the symmetry of the fields are implied by the potential boundary conditions applied to the finite element model. The simplest types of boundary condition are:

Table 6.5 Boundary Conditions

	Field Symmetry	Scalar Potential	Vector Potential
TANGENTIAL MAGNETIC or NORMAL ELECTRIC or PEC	$\mathbf{H} \cdot \mathbf{n} = 0$ $\mathbf{E} \times \mathbf{n} = 0$	$\frac{\partial \phi}{\partial n} = 0$	$\mathbf{A} \times \mathbf{n} = 0$ $V=0$
NORMAL MAGNETIC or TANGENTIAL ELECTRIC	$\mathbf{H} \times \mathbf{n} = 0$ $\mathbf{E} \cdot \mathbf{n} = 0$	$\phi = \text{constant}$	$(\nabla \times \mathbf{A}) \times \mathbf{n} = 0$ $\mathbf{A} \cdot \mathbf{n} = 0$ $\frac{\partial V}{\partial n} = 0$

where \mathbf{n} is the normal unit vector to the surface being considered. Note that in the above ϕ refers to either the reduced or total scalar potential; V refers to the electric scalar potential. A non-zero value for V on an external surface can be used to drive current into a problem. Boundary conditions on the reduced scalar potential only affect the reduced field intensity.

The boundary conditions shown in the table above should only be applied to the exterior surfaces of the finite element model. The default condition that will always be applied if no boundary condition is specified on an exterior surface is:

Table 6.6 Default Boundary Condition

Field Symmetry	Scalar Potential
$\mathbf{H} \cdot \mathbf{n} = 0$	$\frac{\partial \phi}{\partial n} = 0$

The default boundary condition should not be used on vector potential volumes, if the vector region extends to the exterior ‘far field’ boundary. One of the **NORMAL** or **TANGENTIAL** field conditions should be applied.

Nodal Field Extraction

The default boundary condition is only weakly satisfied even when it is assigned to a surface. ‘Weakly satisfied’ means that it is applied as an integral over the surface patch for each shape function sub-domain.

For example, in a scalar potential volume, examination of the field solution close to a surface with the default boundary condition will reveal that the normal component of magnetic field is not zero. The magnitude of the normal component reflects the local accuracy of the solution unless it is the result of a model error. Such an error may arise in a reduced potential region if the conductor's field does not have the same symmetry as the finite element model. On the surface itself however, the normal component is forced to be precisely zero by the post processor.

The $\mathbf{H} \times \mathbf{n}$ boundary condition with a scalar potential and the $\mathbf{H} \cdot \mathbf{n}$ condition with a vector potential solution, are strongly imposed. The potentials are prescribed to the values specified. However, even then the field solution may not be exactly as expected. If the surface normal direction is discontinuous (the normal direction is ambiguous at an edge for example) this will produce a solution which implies at least two possible values for the field at the edge.

The fields computed by taking derivatives of the finite element shape functions will be discontinuous, but in each element the potential boundary condition will be exactly specified. The field smoothing processes in ELEKTRA (nodal averaging, see the notes on accurate field computation in section [“Accurate Fields” on page 6-30](#)) must take this into account, and hence the true boundary condition is forced at these surfaces to ensure accuracy is maintained as far as possible.

Uniqueness

A potential boundary condition must be specified on at least one surface of a magnetic scalar potential problem. This gauges the scalar potential and without it the solution will not be unique. Any constant value could be added over the whole space. In ELEKTRA each independent scalar potential volume must have its own assigned potential surface. This arises where vector potential volumes cut the scalar potential space into distinct, completely separate sub-volumes. An example may help to explain this feature. Consider a long conducting pipe with a short coil inside the pipe. The pipe may be modelled as being infinite, extending between the boundaries of the problem, in which case the interior of the pipe and the exterior will be completely separate volumes that never touch. The scalar potentials in the interior and exterior spaces would need gauging independently. Each one requires at least one assigned potential surface.

The special vector potential formulation used in ELEKTRA does not require the vector or electric scalar potentials to be specified unless they are needed to represent the field symmetry of the problem. The magnitudes of the potentials are automatically gauged by the equation being solved. Most problems will have assigned potentials in fact, but this will be because field symmetries are being specified.

Vector Potential in ELEKTRA

One specific limitation should be mentioned in connection with boundary conditions on the vector potential and its interaction with the algorithm used in ELEKTRA. If a conducting body passes through a plane of symmetry where the flux density is tangential to the plane, the tangential components of the vector potential would be assigned to zero, and the default boundary condition would apply to the normal component of the vector potential. The same geometry and field symmetry would exist if the conductor was cut through at the boundary, but a normal current flow of zero should also be defined. This cannot be done by simply specifying a boundary condition on the normal component of the vector potential, and the following must be applied.

Defining potential boundary conditions on all components of the vector potential would cause all components of the flux to be zero. The cut conductor must be modelled in ELEKTRA by introducing an air volume using a scalar potential between the conductor block and the symmetry surface. This may be inconvenient since the actual cut could be vanishingly thin. It is however essential at the moment.

Voltage driven eddy currents

It is possible to calculate the current flow through a conductor driven by a potential difference between two external surfaces. The entire conductor should be modelled using vector potential volumes. Where the volumes touch the surface of the mesh, the tangential components of \mathbf{A} should be set to zero and the electric scalar potential set to values which define the driving voltage.

Open Boundaries

Electromagnetic fields are frequently not contained within a finite volume. In practice, at long distances from the device producing the field, the distribution will be modified by the local environment, but this will not effect the field close to the device. Except where the interaction with the far environment is of interest, the field from an isolated device is usually required. A simple finite element mesh obviously has a finite extent and applying either potential or derivative boundary conditions on the open boundaries will perturb the true infinite domain solution.

There are techniques that accurately model the infinite domain. On a convex outer surface a series of rings of elements with increasing size may be recursively generated automatically (ballooning) or a boundary integral solution for the exterior domain may be coupled to the interior finite element solution. These techniques are relatively expensive in three dimensions and approximate methods may be used instead. The approximate methods include matching a convex outer surface to finite elements that extend from the surface to infinity and which have appropriate decay functions, and the standard approach of extending the finite element mesh to a distance where the field truncation has no effect on the regions of interest.

It is recommended that the mesh be extended to a reasonable distance with either potential or derivative boundary conditions applied to the outer surface, so that the truncation has an insignificant effect on the region of interest. The effect of truncation can be estimated by observing the tangential component of the field on an open boundary with a derivative boundary condition. Half the field observed on such a boundary is being reflected back from the exterior. Combining this with knowledge of the probable decay in the exterior space will give an order of magnitude for the effect of the truncation on the regions of interest. In a particularly sensitive application a further test should be applied. Two problems should be solved, one using potential boundary conditions on the open boundary, the other using derivative conditions. These problems represent the model in an infinite array of similar problems with either the opposite or the same sign of field in alternate images. In general the two solutions will bound the correct answer. It is often found that the derivative condition is closer to the real solution. It would in fact be exact if the open boundary surface was a constant flux surface. Use of this simple approach gives an estimate for the effect of the far field boundary truncation and this is why it is recommended.

Total and Reduced Potential at the Open Boundary

It is worthwhile considering the significance of the potential type interior to the open boundary surface. With a total potential interior to the open boundary surface both the conductor and iron fields will be reflected at the surface. On the other hand, if a reduced potential is interior to the open surface only the iron fields will be reflected at the surface. This effect may be used to minimise the perturbation caused by the truncation.

For example, if the problem consists of a small volume of iron, interior to a large conductor system, the reduced potential at the open surface will clearly give much smaller errors produced by the approximate open (far field) boundary condition. The relatively small field from the iron will be the only component reflected in the false open boundary.

An alternative example is one where the conductors are contained inside an iron shield. In this case the total field at the open boundary will be much smaller than the conductors' field (if the shield is effective), a total potential next to the open boundary surface will therefore give much less perturbation in the regions of interest.

Accurate Fields

The total and reduced potential formulation used in ELEKTRA allows a problem to be defined with many possible combinations of potentials. The next section looks specifically at the use of total and reduced scalar potentials, and the choices that can be made.

It is assumed firstly that the magnetic vector potential is used where necessary – in eddy current volumes, and in air if required to ensure a singly connected problem. As previously mentioned, it is good practice to use magnetic scalar potentials wherever possible, as this reduces memory requirements, and increases computation speed.

The other restrictions are that the total potential should be used inside magnetic (non-conducting) volumes and the reduced potential inside volumes where source currents are flowing. Apart from these restrictions, the distribution of potential types may be chosen to give optimum accuracy for particular applications. One aspect of this has already been mentioned in section [“Total and Reduced Potential at the Open Boundary” on page 6-30](#)

The following table identifies particular classes of problem, and shows the recommended scalar potential type in a particular volume of space.

In the table, the “iron” field implied induced fields, due to magnetisation and eddy currents. “Nodal” refers to the nodally averaged field output option; “Integral” to the option where the field from the magnetic volumes and eddy current volumes are computed by evaluating the magnetisation from the potential solution, and then computing the field by volume integration of the magnetisation and eddy currents

Table 6.7 Potential Types

Problem Description	Potential Type	Field Recovery	
		from mesh	from conductors
Homogeneous Field Conductor field » Iron field Close to the iron Far from the iron	Reduced Any	Nodal Integral	Integral Integral
Homogeneous Field Conductor Field \approx Iron Field Homogeneous conductor field Inhomogeneous conductor field or if total cannot be used	Any Total Reduced	Nodal Nodal Nodal	Integral Integral Integral
Homogeneous Field Conductor Field \approx -Iron Field Homogeneous conductor field Inhomogeneous conductor field or if total cannot be used	Total Total Reduced	Nodal Nodal Nodal	Any Any Integral
Homogeneous Field Conductor Field « Iron Field	Any	Nodal	Integral
Peak Field on Conductor	Reduced	Nodal	Integral
Inhomogeneous Field Conductor Field » Iron field Conductor Field « Iron field As above but far from iron Conductor Field \approx Iron field Conductor Field \approx -Iron field	Reduced Any Any Any Total	Nodal Nodal Integral Nodal Nodal	Integral Integral Integral Integral Integral

The table does not cover all eventualities and it may help to explain why the combinations are recommended. First, the use of total potential in volumes where the source and iron fields cancel is the basis on which the total and reduced potential algorithm was designed. This takes precedence over all other considerations.

Finite elements are good at representing uniform fields and by implication, not so good for varying fields. If the combined source and iron fields produce a uniform resultant field, then it is best to use the total potential in such volumes. However, if the source field is much greater than the iron field, it is best to use a reduced potential in this volume because the source fields are essentially exact.

When ELEKTRA results are displayed using the OPERA-3d post processor, the effects of the various types of field evaluation can be tested. The error caused by

using interpolated source fields can be immediately compared against the more accurate but expensive source integration procedure. It is recommended that when peak fields on coils or body forces on coils are required, then source fields should be calculated by integration.

The fields computed by magnetisation integration are a very specialised option. The primary role of this option is in applications such as shielded superconducting NMR systems. Field accuracy of the order of parts in 10^6 are required at some distance from the shield and where the shield contributes less than 10% of the total field. Reduced potential could be used in these volumes, providing the conductor fields are uniform. This can achieve an accuracy of 3 in 10^5 . Comparison with analytical solutions has showed that the magnetisation integration can improve this to 3 in 10^6 . However this procedure is very computationally expensive.

Magnetic Shielding

Many of the recommendations in the sections on accuracy apply to the calculation of electromagnetic fields in applications where shielding is the primary objective. The most important recommendations for shielding are that in the volumes where the shield is effective it is essential that the total potential option is used and fields are evaluated either from the shape function derivatives, or better still from the nodally averaged field.

When the shield is screening the outside world from the field produced by a device the above recommendation means that a total potential will be used in the exterior space. This is also important from the point of the approximate far field boundary condition that will be imposed at the boundary of the mesh. The total potential gives minimal perturbation of the interior solution.

Thin magnetic screens are often used at large distances from the device to be isolated. Elements with large aspect ratios can be used in such screens and will not adversely affect the solution accuracy. However, if a detailed evaluation of the field at a corner or next to a small gap is required, then the elements' dimension must be small compared to the feature being modelled. The restriction is perfectly reasonable, in any direction where the field is changing rapidly the element discretisation must be capable of modelling the changes.

Velocity Induced Eddy Currents in ELEKTRA-VL

The solution of motion induced eddy current problems (without upwinding) will probably present oscillations if the 'cell Peclet number' $\mu\sigma\mathbf{u}h$ is much greater than one (h being the size of the element in the velocity direction \mathbf{u}). This still may

occur after mesh refinement, and especially if quadratic elements are used. Such a phenomenon is well understood and is typical of the numerical methods. It arises when first order derivatives become very large compared to second order ones (see motional equations). To overcome this problem ‘upwinding’ can be employed to eliminate non-physical ‘wiggles’ from the solution although with a possible degradation in accuracy. The number of iterations required by the solver is usually smaller than that for non-upwinded solutions, since the system matrix is typically better conditioned. Such a technique (e.g. Hughes upwinding) can only be combined with linear elements.

Permeable Vector Potential Volumes

For some types of problem, it is possible to improve the solution time considerably by careful choice of where volumes of differing potential type are placed. A particular instance is when highly permeable vector potential volumes are placed adjacent to magnetic scalar air volumes.

To improve the solution time for this situation, it is recommended that the vector potential volume be extended into the air for a short distance. In this way, the jump from vector to scalar potential does not coincide with a large change in permeability.

CARMEN, Rotating Machines Analysis

Introduction

This section describes the electromagnetic field analysis program CARMEN, which is one of the analysis programs of the OPERA-3d Vector Fields Electromagnetic Analysis Environment.

CARMEN can be used to compute electromagnetic fields including the effects of eddy currents for rotating systems in three dimensions. The program incorporates state of the art algorithms for the calculation of electromagnetic fields and advanced finite element numerical analysis procedures. In the following sections the algorithm used in CARMEN is described so that users are able to relate the finite element model to their application problem.

The CARMEN Algorithm

CARMEN can use a combination of vector and scalar magnetic potentials to model time varying electromagnetic fields. Vector potentials have to be used in conducting media; scalar potentials can be used in the rest of space (with certain restrictions that depend on the geometry).

Fields in Source Free, Non-Conducting Media

In a region of free space that does not include source currents, it is generally found to be most efficient to replace the magnetic field by the gradient of a magnetic scalar potential, ψ , so that:

$$\mathbf{H} = -\nabla\psi \quad (6.52)$$

In a similar way to the TOSCA algorithm outlined for electrostatic problems in section “TOTAL and REDUCED Magnetic Scalar Potentials” on page 6-9, this leads to a governing equation:

$$\nabla \cdot \mu \nabla \psi = 0 \quad (6.53)$$

where $\mathbf{B} = \mu\mathbf{H}$.

Fields in Non-Conducting Media With Sources

Fields in non-conducting media with sources in general consist of both solenoidal and rotational components. The field produced by known electric (source) currents has a rotational component inside the volumes where the currents flow. In the rest of free space the field is solenoidal but the scalar potential is multi-valued. The field produced by magnetized volumes is solenoidal.

It is convenient to use a splitting of the total field into two parts in order to obtain a description of the field in terms of a simple scalar potential in non-conducting media. The total field intensity \mathbf{H} is defined using the reduced field intensity ($\mathbf{H}_m + \mathbf{H}_e$) and the conductor field intensity (\mathbf{H}_s):

$$\mathbf{H} = \mathbf{H}_m + \mathbf{H}_e + \mathbf{H}_s \quad (6.54)$$

The reduced field intensity has two contributions, one from magnetisation fields in iron (\mathbf{H}_m), and the other due to fields from induced eddy currents in conductors (\mathbf{H}_e). The sum of these two is the unknown quantity, and can now be represented using the reduced scalar potential (ϕ):

$$\mathbf{H}_m + \mathbf{H}_e = -\nabla\phi \quad (6.55)$$

In the case of stationary magnetic fields where the exciting currents are prescribed, the conductor field intensity may always be directly evaluated by integration:

$$\mathbf{H}_s = \int_{\Omega_J} \frac{\mathbf{J} \times \mathbf{R}}{|\mathbf{R}|^3} d\Omega_J \quad (6.56)$$

For further details of the use of total and reduced scalar potentials in non-conducting media, look at section [“TOTAL and REDUCED Magnetic Scalar Potentials” on page 6-9](#).

Fields in Conducting Media

In time varying fields or rotating systems, the currents that are induced in conducting volumes are part of the unknowns in the system. Their fields cannot therefore be evaluated by simply performing an integration. Inside the conducting volumes the field representation must include a rotational component. The most elegant approach is to combine the efficient total and reduced scalar potential method for non-conducting volumes with an algorithm that uses a vector potential (\mathbf{A}) in the conducting volumes.

In a rotating system, where the conductors are moving, or the applied field is rotating, the magnetic and electric fields are related by the low frequency limit of Maxwell's equations:

$$\nabla \times \mathbf{H} = \mathbf{J} \quad (6.57)$$

$$\nabla \times \mathbf{E} = -\frac{\partial \mathbf{B}}{\partial t} \quad (6.58)$$

where

$$\mathbf{J} = \sigma(\mathbf{E} + \mathbf{u} \times \mathbf{B}) \quad (6.59)$$

and σ is the electrical conductivity and \mathbf{u} the velocity. At this point we assume that the velocity is only rotational.

The restriction encountered with ELEKTRA-VL was that in order to include the velocity term, the geometry must be unchanging in the direction of motion. This restriction is removed here. It is necessary therefore to solve the rotation as a transient phenomenon, requiring a fully transient analysis.

The eddy currents induced by motion are implied by the following equations, along with the fact that at each time step, the rotating part of the model is physically moved. Including the motion explicitly in this way means the velocity term is not required in the following equations.

The flux density \mathbf{B} can be represented using a vector potential \mathbf{A} where:

$$\mathbf{B} = \nabla \times \mathbf{A} \quad (6.60)$$

Using this vector potential and combining equations 6.33 to 6.36 gives the following equation for \mathbf{A} :

$$\nabla \times \frac{1}{\mu} \nabla \times \mathbf{A} = -\sigma \frac{\partial \mathbf{A}}{\partial t} - \sigma \nabla V \quad (6.61)$$

The electric scalar potential (V) emerges because of the non-uniqueness of the potential which arises during integration of equation 6.33.

To ensure a unique solution therefore, it is also necessary to ‘gauge’ equation 6.37. The gauge used in CARMEN is the Coulomb gauge

$$\nabla \cdot \mathbf{A} = 0 \quad (6.62)$$

and after including it into equation 6.37 the new governing equation is

$$\nabla \times \frac{1}{\mu} \nabla \times \mathbf{A} - \nabla \frac{1}{\mu} \nabla \cdot \mathbf{A} + \sigma \left(\frac{\partial \mathbf{A}}{\partial t} + \nabla V \right) = 0 \quad (6.63)$$

It is also necessary to solve the secondary equation

$$\nabla \cdot \sigma \nabla V + \nabla \cdot \sigma \frac{\partial \mathbf{A}}{\partial t} = 0 \quad (6.64)$$

This vector potential description of fields inside conducting volumes can be directly combined with the total and reduced potential description in non-conducting spaces. The normal flux and tangential field intensity interface conditions are used to exactly couple the two field descriptions.

Including Rotation

In CARMEN, part of the mesh is allowed to rotate about the z-axis, and this rotation may also induce eddy currents in conducting media. The technique used to model this is called the “lock-step” algorithm. The element subdivision around a slip surface must be uniform in the direction of rotation, and the rotating part of the mesh moves one element each time step. In this way, the mesh on either side of the slip surface remains continuous at all times. The angle subtended at the origin for one element, along with the rotational velocity, is used internally to define the time step to be used by the transient solution.

It is essential therefore that to ensure a regular subdivision on the slip surface, that hexahedral elements only are used. It is not possible to have correct continuity of the mesh if tetrahedra are used.

The slip surface is defined as a boundary condition, and runs the entire length of the device (z-direction). It should have a constant radius the entire length, and is labelled **SLIP** in the boundary definition stage of the pre processor. As an alternative to selecting the slip surface in the boundary definition stage, the **SLIP** boundary condition can be applied using the OPERA-3d pre processor **SLIP** command (accessed through **Add slip surface** on the **MODIFY** menu). The user specifies the radius of the slip surface and the tolerance for inclusion of points not precisely at this radius.

It is a requirement that the slip surface **ONLY** be within total magnetic scalar potential volumes, and that vector potentials or reduced scalar potentials **DO NOT** touch the slip surface. This is to ensure as efficient a solution system as possible (by allowing a rotating coordinate system in the rotating part of the mesh – not possible if vector potential or reduced potential volumes touch the slip surface!).

The intermediate points at which solutions are to be stored are defined by the angles to which the rotating mesh has moved (unlike ELEKTRA-TR where these points are defined in terms of times elapsed).

Time Variation in CARMEN

The Transient Analysis Program for rotating systems (CARMEN) solves eddy current problems where the driving currents are changing in time in a predetermined way, but not necessarily with a time harmonic behaviour, and where the conducting materials may also be rotating.

If the Galerkin procedure is applied to the terms in \mathbf{A} and $\frac{\partial \mathbf{A}}{\partial t}$ in equation 6.16 both yield matrices, referred to as \mathbf{R} and \mathbf{S} . This leads to a matrix equation

$$\mathbf{R}\mathbf{A} + \mathbf{S}\frac{\partial \mathbf{A}}{\partial t} + \mathbf{B} = 0 \quad (6.65)$$

where \mathbf{A} is now a vector of unknown potentials and \mathbf{B} is a vector of driving terms. The solution of equation 6.44 is also based on the Galerkin procedure. \mathbf{A} and \mathbf{B} are discretised in time using a first order function of time:

$$\mathbf{A}(t) = (1 - \tau)a_n + \tau a_{n+1} \quad (6.66)$$

$$\mathbf{B}(t) = (1 - \tau)b_n + \tau b_{n+1} \quad (6.67)$$

where

$$\tau = \frac{t - t_n}{t_{n+1} - t_n} \quad (6.68)$$

and a_n and b_n are values of \mathbf{A} and \mathbf{B} at time t_n . Using τ as the weight in a Galerkin weighted residual solution of 6.44 leads to a recurrence relationship between a_{n+1} and a_n :

$$\left(\mathbf{R}(1 - \theta) - \frac{\mathbf{S}}{\Delta t}\right)a_n + \left(\mathbf{R}\theta + \frac{\mathbf{S}}{\Delta t}\right)a_{n+1} + b_n(1 - \theta) + b_{n+1}\theta = 0 \quad (6.69)$$

where $\theta = 1 - \tau$ and the time step $\Delta t = t_{n+1} - t_n$.

The time step is determined from the speed of rotation, and the azimuthal subdivision on the slip surface. Since the rotating part of the mesh moves one element on the slip surface per time step, the actual value of time step is beyond the control of the user, and is fully defined by the software.

Transient Driving Functions

The ‘driving’ field is provided by source currents, J_s , or non-zero potential boundary conditions. The available drive functions within CARMEN are as follows:

Table 6.8 Transient Drive Functions

Name	Function
DC	$F=1$ for all time
STEP	$t < 0: F=0$ $t \geq 0: F=1$
RAMP	$t < 0: F=0$ $0 \leq t \leq t_c: F = \frac{t}{t_c}$ $t > t_c: F=1.$
SINE	$t < 0: F=F(0)$ $t \geq 0: F=\sin 2\pi ft + \phi.$
COSINE	$t < 0: F=F(0)$ $t \geq 0: F=\cos 2\pi ft + \phi.$
PEAK	$t < 0: F=0$ $t \geq 0: F = a \exp(-t^2/b)$ a and b are chosen such that $F=1$ at $t=t_c.$
RISE	$t < 0: F=0$ $t \geq 0: F = 1 - \exp(-t/t_c)$
TABLE (switch on)	$t < 0: F=0$ $t \geq 0: F=\text{cubic splines}.$
TOFF (switch off)	$t < 0: F=F(0)$ where $F(0)$ is the value of the function in the table file at time $t=0$. $t \geq 0: F=\text{cubic splines}.$

The driving function can be applied to all the source conductors and boundary conditions, or the drive label can be used to specify different driving functions for different conductors and boundary conditions. A phase angle can also be used in the sine and cosine functions.

The transient time table option allows the user to define driving functions other than those programmed into the analysis code. The tables consist of files containing pairs of numbers in free format, one pair per line. The first number on each line specifies the time; the second gives the function value. A total of 4096 pairs of points are allowed, distributed arbitrarily between the time table files. To check the shape of the drive function, the **GRAPH** command can be used to display the file contents.

For the time table **TABLE** “Switch On” option, the values of time should start at zero and increase through the file. Discontinuities in function value or first derivative can be forced by specifying two entries for the same value of time. For the time table **TOFF** “Switch Off” option, the value of the function is set to the value of the drive function in the time table at $t=0$ for all time up to time zero.

Other functions (**DC**, **COSINE**, and **SINE**) will also have function values that are non-zero at time zero (depending on the phase angle for sine and cosine drives).

Beyond the last value of time in the table, the function continues with the last value computed.

Non-Linear Materials

The analysis program CARMEN now includes non-linear material properties, which can also be anisotropic. Permanent magnets can be included, and the behaviour of these materials is also defined by BH curves that extend into the 3rd quadrant.

In each case, a direct iteration method is used (rather than Newton Raphson used in TOSCA). This implies that in general more iterations will be required to converge to a non-linear solution.

Boundary Conditions

Boundary conditions are used in two ways. Firstly they can provide a way of reducing the size of the finite element representation of symmetrical problems. Secondly they are used to approximate the magnetic field at large distances from the problem (far-field boundaries).

General

Problem symmetry and the symmetry of the fields are implied by the potential boundary conditions applied to the finite element model. The simplest types of boundary condition are:

Table 6.9 Boundary Conditions

	Field Symmetry	Scalar Potential	Vector Potential
TANGENTIAL MAGNETIC or NORMAL ELECTRIC or PEC	$\mathbf{H} \cdot \mathbf{n} = 0$ $\mathbf{E} \times \mathbf{n} = 0$	$\frac{\partial \phi}{\partial n} = 0$	$\mathbf{A} \times \mathbf{n} = 0$ $V=0$
NORMAL MAGNETIC or TANGENTIAL ELECTRIC	$\mathbf{H} \times \mathbf{n} = 0$ $\mathbf{E} \cdot \mathbf{n} = 0$	$\phi = \text{constant}$	$(\nabla \times \mathbf{A}) \times \mathbf{n} = 0$ $\mathbf{A} \cdot \mathbf{n} = 0$ $\frac{\partial V}{\partial n} = 0$

where \mathbf{n} is the normal unit vector to the surface being considered. Note that in the above ϕ refers to either the reduced or total scalar potential; V refers to the electric scalar potential. A non-zero value for V on an external surface can be used to drive current into a problem. Boundary conditions on the reduced scalar potential only affect the reduced field intensity.

The boundary conditions shown in the table above should only be applied to the exterior surfaces of the finite element model. The default condition that will always be applied if no boundary condition is specified on an exterior surface is:

Table 6.10 Default Boundary Condition

Field Symmetry	Scalar Potential
$\mathbf{H} \cdot \mathbf{n} = 0$	$\frac{\partial \phi}{\partial n} = 0$

The default boundary condition should not be used on vector potential volumes, if the vector region extends to the exterior ‘far field’ boundary. One of the **NORMAL** or **TANGENTIAL** field conditions should be applied.

Nodal Field Extraction

The default boundary condition is only weakly satisfied even when it is assigned to a surface. ‘Weakly satisfied’ means that it is applied as an integral over the surface patch for each shape function sub-domain.

For example, in a scalar potential volume, examination of the field solution close to a surface with the default boundary condition will reveal that the normal component of magnetic field is not zero. The magnitude of the normal component reflects the local accuracy of the solution unless it is the result of a model error. Such an error may arise in a reduced potential region if the conductor's field does not have the same symmetry as the finite element model. On the surface itself however, the normal component is forced to be precisely zero by the post processor.

The $\mathbf{H} \times \mathbf{n}$ boundary condition with a scalar potential and the $\mathbf{H} \cdot \mathbf{n}$ condition with a vector potential solution, are strongly imposed. The potentials are prescribed to the values specified. However, even then the field solution may not be exactly as expected. If the surface normal direction is discontinuous (the normal direction is ambiguous at an edge for example) this will produce a solution which implies at least two possible values for the field at the edge.

The fields computed by taking derivatives of the finite element shape functions will be discontinuous, but in each element the potential boundary condition will be exactly specified. The field smoothing processes in CARMEN (nodal averaging, see the notes on accurate field computation in section “[Accurate Fields](#)” on [page 6-45](#)) must take this into account, and hence the true boundary condition is forced at these surfaces to ensure accuracy is maintained as far as possible.

Uniqueness

A potential boundary condition must be specified on at least one surface of a magnetic scalar potential problem. This gauges the scalar potential and without it the solution will not be unique. Any constant value could be added over the whole space. In CARMEN each independent scalar potential volume must have its own assigned potential surface. This arises where vector potential volumes cut the scalar potential space into distinct, completely separate sub-volumes.

An example may help to explain this feature. Consider a long conducting pipe with a short coil inside the pipe. The pipe may be modelled as being infinite, extending between the boundaries of the problem, in which case the interior of the pipe and the exterior will be completely separate volumes that never touch. The scalar potentials in the interior and exterior spaces would need gauging independently. Each one requires at least one assigned potential surface.

The special vector potential formulation used in CARMEN does not require the vector or electric scalar potentials to be specified unless they are needed to represent the field symmetry of the problem. The magnitudes of the potentials are automatically gauged by the equation being solved. Most problems will have assigned potentials in fact, but this will be because field symmetries are being specified.

Vector Potential in CARMEN

One specific limitation should be mentioned in connection with boundary conditions on the vector potential and its interaction with the algorithm used in CARMEN. If a conducting body passes through a plane of symmetry where the flux density is tangential to the plane, the tangential components of the vector potential would be assigned to zero, and the default boundary condition would apply to the normal component of the vector potential. The same geometry and field symmetry would exist if the conductor was cut through at the boundary, but a normal current flow of zero should also be defined. This cannot be done by simply specifying a boundary condition on the normal component of the vector potential.

Defining potential boundary conditions on all components of the vector potential would cause all components of the flux to be zero. The cut conductor must be modelled in CARMEN by introducing an air volume using a scalar potential between the conductor block and the symmetry surface. This may be inconvenient since the actual cut could be vanishingly thin. It is however essential at the moment.

Open Boundaries

Electromagnetic fields are frequently not contained within a finite volume. In practice, at long distances from the device producing the field, the distribution will be modified by the local environment, but this will not effect the field close to the device. Except where the interaction with the far environment is of interest, the field from an isolated device is usually required. A simple finite element mesh obviously has a finite extent and applying either potential or derivative boundary conditions on the open boundaries will perturb the true infinite domain solution.

There are techniques that accurately model the infinite domain. On a convex outer surface a series of rings of elements with increasing size may be recursively generated automatically (ballooning) or a boundary integral solution for the exterior domain may be coupled to the interior finite element solution. These techniques are relatively expensive in three dimensions and approximate methods may be used instead. The approximate methods include matching a convex outer surface to finite elements that extend from the surface to infinity and which have appropriate decay functions, and the standard approach of extending the finite element mesh to a distance where the field truncation has no effect on the regions of interest.

It is recommended that the mesh be extended to a reasonable distance with either potential or derivative boundary conditions applied to the outer surface, so that the truncation has an insignificant effect on the region of interest. The effect of truncation can be estimated by observing the tangential component of the field on an open boundary with a derivative boundary condition. Half the field observed on such a boundary is being reflected back from the exterior. Combining this with knowledge of the probable decay in the exterior space will give an order of mag-

nitude for the effect of the truncation on the regions of interest. In a particularly sensitive application a further test should be applied. Two problems should be solved, one using potential boundary conditions on the open boundary, the other using derivative conditions. These problems represent the model in an infinite array of similar problems with either the opposite or the same sign of field in alternate images. In general the two solutions will bound the correct answer. It is often found that the derivative condition is closer to the real solution. It would in fact be exact if the open boundary surface was a constant flux surface. Use of this simple approach gives an estimate for the effect of the far field boundary truncation and this is why it is recommended.

Total and Reduced Potential at the Open Boundary

It is worthwhile considering the significance of the potential type interior to the open boundary surface. With a total potential interior to the open boundary surface both the conductor and iron fields will be reflected at the surface. On the other hand, if a reduced potential is interior to the open surface only the iron fields will be reflected at the surface. This effect may be used to minimise the perturbation caused by the truncation.

For example, if the problem consists of a small volume of iron, interior to a large conductor system, the reduced potential at the open surface will clearly give much smaller errors produced by the approximate open (far field) boundary condition. The relatively small field from the iron will be the only component reflected in the false open boundary.

An alternative example is one where the conductors are contained inside an iron shield. In this case the total field at the open boundary will be much smaller than the conductors' field (if the shield is effective), a total potential next to the open boundary surface will therefore give much less perturbation in the regions of interest.

Accurate Fields

The total and reduced potential formulation used in CARMEN allows a problem to be defined with many possible combinations of potentials. The next section looks specifically at the use of total and reduced scalar potentials, and the choices that can be made.

It is assumed firstly that the magnetic vector potential is used where necessary – in eddy current volumes, and in air if required to ensure a singly connected problem. As previously mentioned, it is good practice to use magnetic scalar potentials wherever possible, as this reduces memory requirements, and increases computation speed.

The other restrictions are that the total potential should be used inside magnetic (non-conducting) volumes, on either side of a slip surface, and the reduced potential must be used inside volumes where source currents are flowing. Apart from these restrictions, the distribution of potential types may be chosen to give optimum accuracy for particular applications. One aspect of this has already been mentioned in section [“Total and Reduced Potential at the Open Boundary”](#) on [page 6-45](#).

The following table identifies particular classes of problem, and shows the recommended scalar potential type in a particular volume of space.

In the table, the “iron” field implied induced fields, due to magnetisation and eddy currents. “Nodal” refers to the nodally averaged field output option; “Integral” to the option where the field from the magnetic volumes and eddy current volumes are computed by evaluating the magnetisation from the potential solution, and then computing the field by volume integration of the magnetisation and eddy currents.

Table 6.11 Potential Types

Problem Description	Potential Type	Field Recovery	
		from mesh	from conductors
Homogeneous Field Conductor field » Iron field Close to the iron Far from the iron	Reduced Any	Nodal Integral	Integral Integral
Homogeneous Field Conductor Field \approx Iron Field Homogeneous conductor field Inhomogeneous conductor field or if total cannot be used	Any Total Reduced	Nodal Nodal Nodal	Integral Integral Integral
Homogeneous Field Conductor Field \approx -Iron Field Homogeneous conductor field Inhomogeneous conductor field or if total cannot be used	Total Total Reduced	Nodal Nodal Nodal	Any Any Integral
Homogeneous Field Conductor Field « Iron Field	Any	Nodal	Integral
Peak Field on Conductor	Reduced	Nodal	Integral
Inhomogeneous Field Conductor Field » Iron field Conductor Field « Iron field As above but far from iron Conductor Field \approx Iron field Conductor Field \approx -Iron field	Reduced Any Any Any Total	Nodal Nodal Integral Nodal Nodal	Integral Integral Integral Integral Integral

The table does not cover all eventualities and it may help to explain why the combinations are recommended. First, the use of total potential in volumes where the source and iron fields cancel is the basis on which the total and reduced potential algorithm was designed. This takes precedence over all other considerations.

Finite elements are good at representing uniform fields and by implication, not so good for varying fields. If the combined source and iron fields produce a uniform resultant field, then it is best to use the total potential in such volumes. However, if the source field is much greater than the iron field, it is best to use a reduced potential in this volume because the source fields are essentially exact.

When CARMEN results are displayed using the OPERA-3d post processor, the effects of the various types of field evaluation can be tested. The error caused by

using interpolated source fields can be immediately compared against the more accurate but expensive source integration procedure. It is recommended that when peak fields on coils or body forces on coils are required, then source fields should be calculated by integration.

Magnetic Shielding

Many of the recommendations in the sections on accuracy apply to the calculation of electromagnetic fields in applications where shielding is the primary objective. The most important recommendations for shielding are that in the volumes where the shield is effective it is essential that the total potential option is used and fields are evaluated either from the shape function derivatives, or better still from the nodally averaged field.

When the shield is screening the outside world from the field produced by a device the above recommendation means that a total potential will be used in the exterior space. This is also important from the point of the approximate far field boundary condition that will be imposed at the boundary of the mesh. The total potential gives minimal perturbation of the interior solution.

Thin magnetic screens are often used at large distances from the device to be isolated. Elements with large aspect ratios can be used in such screens and will not adversely affect the solution accuracy. However, if a detailed evaluation of the field at a corner or next to a small gap is required, then the elements' dimension must be small compared to the feature being modelled. The restriction is perfectly reasonable, in any direction where the field is changing rapidly the element discretisation must be capable of modelling the changes.

Permeable Vector Potential Volumes

For some types of problem, it is possible to improve the solution time considerably by careful choice of where volumes of differing potential type are placed. A particular instance is when highly permeable vector potential volumes are placed adjacent to magnetic scalar air volumes.

To improve the solution time for this situation, it is recommended that the vector potential volume be extended into the air for a short distance. In this way, the jump from vector to scalar potential does not coincide with a large change in permeability.

SOPRANO, High Frequency Analysis

Introduction

This section describes the electromagnetic field analysis program SOPRANO, which is one of the analysis programs of the OPERA-3d Vector Fields Electromagnetic Analysis Environment.

SOPRANO can be used to compute high frequency electromagnetic fields, including eigenvalues, in three dimensions. The program incorporates state of the art algorithms for the calculation of electromagnetic fields and advanced finite element numerical analysis procedures. In the following sections the algorithm used in SOPRANO is described so that users are able to relate the finite element model to their application problem.

The SOPRANO Algorithm

The Eigen Value Solver

The Eigenvalue solver, SOPRANO-EV, is designed to solve resonant cavity problems containing lossless, isotropic dielectrics. The walls of a cavity are treated as perfect electrical conductors (PEC) which is a good approximation at high frequency.

SOPRANO-EV models the electric field directly using “edge” elements rather than using the potential formulation used by ELEKTRA. In this approach the degrees of freedom associated with \mathbf{E} are not its components directly, but the emf jump between connected nodes. The emf $e_{\{ij\}}$ is defined as,

$$e_{\{ij\}} = \int_{l_{\{ij\}}} \mathbf{E} \cdot d\mathbf{l} \quad (6.70)$$

where i and j are two nodes in the mesh and $l_{\{ij\}}$ is the *directed* edge connecting them. This is the origin of the term “edge” in edge elements. In some respects this approach is very similar to the method of Yee in Finite Difference Time Domain. The advantage of edge elements is that they model the important singular field behaviour around re-entrant corners accurately and in a very natural manner.

At high frequency the Electric field obeys the vector wave equation:

$$\nabla \times \frac{1}{\mu} \nabla \times \mathbf{E} = -\frac{\partial^2 (\epsilon \mathbf{E})}{\partial t^2} \quad (6.71)$$

with the following condition holding on **PEC** boundaries

$$\mathbf{E} \times \mathbf{n} = 0 \quad (6.72)$$

Further assuming all materials within the cavity are linear, (6.71) reduces to the vector Helmholtz equation for each individual mode m ,

$$\nabla \times \frac{1}{\mu} \nabla \times \mathbf{E} - \epsilon \omega_m^2 \mathbf{E} = 0 \quad (6.73)$$

Applying the Galerkin method as usual, this leads to a sparse generalised eigenvalue problem,

$$\mathbf{K}\mathbf{e} = \omega_m^2 \mathbf{M}\mathbf{e} \quad (6.74)$$

from which the eigenvalues ω_m can be determined.

The Time Harmonic Solver

SOPRANO-SS also uses edge elements to solve the wave equation in terms of the electric field. Prior to version 7.5, a nodal formulation in terms of the magnetic vector and electric scalar potentials was used, and for compatibility boundary conditions are still defined in terms of magnetic vector potential. The equation solved is identical to the equation for SOPRANO-EV except that the angular frequency ω is defined by the user i.e. a deterministic solution of the equation is made.

$$\nabla \times \frac{1}{\mu} \nabla \times \mathbf{E} - \epsilon \omega^2 \mathbf{E} = 0 \quad (6.75)$$

Again, the assumption is that all materials are linear.

Time Variation in SOPRANO

There are two SOPRANO analysis modules, each solving a different form of time variation.

- SOPRANO-SS calculates steady-state fields (the time harmonic form) where all fields and potentials are oscillating at the same frequency.
- SOPRANO-EV calculates the modes of an oscillating system, returning the characteristic frequencies (the eigenvalues) and related modes of vibration (eigenvectors).

The Time Harmonic Equations

Under steady state alternating excitation a complex potential substitution can be used. This carries the assumption that the materials have linear characteristics and that therefore the waveforms of the potential and field are precisely the same. The substitutions are:

$$\begin{aligned} E(t) &= E_c e^{i\omega t} \\ H(t) &= H_c e^{i\omega t} \end{aligned} \quad (6.76)$$

where ω is the angular frequency of the alternating excitation.

For high frequency analyses, it is not usual to use conductivity, but instead to use a complex permittivity, where the imaginary part takes into account the losses of any material, $\epsilon = \epsilon' - i\epsilon''$. The conductivity is then equivalent to the term $\omega\epsilon''$.

With these assumption equation 6.75 becomes

$$\nabla \times \frac{1}{\mu} \nabla \times \mathbf{E}_c - \omega^2 \epsilon' \mathbf{E}_c + i\omega^2 \epsilon'' \mathbf{E}_c = 0 \quad (6.77)$$

or

$$\nabla \times \frac{1}{\mu} \nabla \times \mathbf{E}_c - \omega^2 \epsilon' \mathbf{E}_c + i\omega \sigma \mathbf{E}_c = 0 \quad (6.78)$$

The user may specify either the permittivity and the conductivity or the modulus and phase of the complex permittivity.

The Eigenvalue Solution

The Eigenvalue Program (SOPRANO-EV) finds the eigenvalues closest to a given frequency, or within a specified range. In the pre processor the user selects which method is preferred, and is prompted for values accordingly.

The method to find a range of eigenvalues must be to used with some care. As a rule of thumb the number of eigenvalues in a unit range about the n^{th} eigenvalue is proportional to n^3 (this is a property of three dimensional systems). For example if the first two eigenvalues of a system are 1 and 2, then approximately 8 eigenvalues fall in the range 1-3 and 27 in the range 1-4.

The Eigenvalue solver must hold all eigenvectors in memory if it is to determine all the values in a given range. As computer resources are limited the eigenvalue solver will be able to store only a limited number of solutions. Therefore it is advised that the range of the search in the eigenvalue spectrum is kept as small as

possible. Preferably run repeatedly a small model to sketch out the general spectrum of eigenvalues, and then a larger model to determine the values of interest to a higher level of accuracy, searching for the eigenvalues closest to a given frequency.

Boundary Conditions

Boundary conditions are used in two ways. Firstly they can provide a way of reducing the size of the finite element representation of symmetrical problems. Secondly they are used to approximate the magnetic field at large distances from the problem (far-field boundaries).

General

Problem symmetry and the symmetry of the fields are implied by the potential boundary conditions applied to the finite element model. The simplest types of boundary condition are:

Table 6.12 Boundary Conditions

	Field Symmetry	Vector Potential
TANGENTIAL MAGNETIC or NORMAL ELECTRIC or PEC	$\mathbf{H} \cdot \mathbf{n} = 0$ $\mathbf{E} \times \mathbf{n} = 0$	$\mathbf{A} \times \mathbf{n} = 0$ $V=0$
NORMAL MAGNETIC or TANGENTIAL ELECTRIC	$\mathbf{H} \times \mathbf{n} = 0$ $\mathbf{E} \cdot \mathbf{n} = 0$	$(\nabla \times \mathbf{A}) \times \mathbf{n} = 0$ $\mathbf{A} \cdot \mathbf{n} = 0$ $\frac{\partial V}{\partial n} = 0$
RADIATION	See section “Open Boundaries” on page 6-53.	

where \mathbf{n} is the normal unit vector to the surface being considered. Note that in the above V refers to the electric scalar potential.

One of the **NORMAL** or **TANGENTIAL** field conditions or the **RADIATION** condition should be applied on all exterior boundaries.

Sources in SOPRANO-SS, the deterministic steady state solution, are defined by assigning non-zero values to components of the vector potential boundary conditions. (See “The Time Harmonic Solver” on page 6-50.) Since

$\mathbf{E} = -\frac{\partial \mathbf{A}}{\partial t} = -i\omega \mathbf{A}$ (assuming $\nabla V = 0$), specifying values of \mathbf{A} on the port of a microwave structure also specifies \mathbf{E} . For example, specifying

$$A_x = A_z = 0, A_y = \frac{1 \times 10^{-9}}{2\pi} \cos\left(\frac{\pi x}{0.086}\right) \quad (6.79)$$

on the boundary representing a port of a microwave device 86 mm wide in X, centred on X = 0 on the XY plane and operating at 1GHz applies the fundamental mode of the Y-directed electric field with peak value of 1 V/m.

Nodal Field Extraction

The default boundary condition is only weakly satisfied even when it is assigned to a surface. ‘Weakly satisfied’ means that it is applied as an integral over the surface patch for each shape function sub-domain.

For example, examination of the magnetic field solution close to a PEC surface will reveal that the normal component of the field is not zero. The magnitude of the normal component reflects the local accuracy of the solution unless it is the result of a model error. On the surface itself however, the normal component is forced to be precisely zero by the post processor.

Open Boundaries

Electromagnetic fields are frequently not contained within a finite volume. In practice, at long distances from the device producing the field, the distribution will be modified by the local environment, but this will not effect the field close to the device. Except where the interaction with the far environment is of interest, the field from an isolated device is usually required. A simple finite element mesh obviously has a finite extent and applying either potential or derivative boundary conditions on the open boundaries will perturb the true infinite domain solution, and will in fact reflect all energy incident upon it.

There are techniques that accurately model the infinite domain, and one of these has been implemented in SOPRANO-SS. It is only necessary to identify the outer boundary using the boundary condition **RADIATION**. This imposes an Absorbing Boundary Condition, which effectively absorbs any energy incident upon it, and does not reflect the energy back onto the model being solved.

It is recommended that the mesh be extended to a reasonable distance, typically about 1 wavelength, using at least 10-15 elements per wavelength. The outer

boundary should be convex, without sharp corners. Better results will be obtained the closer this boundary is to spherical in shape.

Ideally the radiating source in the problem should be at the approximate centre of the coordinate system. The absorbing boundary is then a sphere, centred on the origin of the coordinate system, giving the best accuracy. In general, take care not to place the boundary at or near the origin, as this can cause excessive errors.

Eigenvalue Boundary Conditions

Within SOPRANO-EV, there are no sources included in the model – it is not a driven problem. Other than symmetry planes, the model must be enclosed by perfect electric walls, with no **RADIATION** conditions present. Symmetry may still be exploited in the usual way, as long as it implies a closed volume that is to be modelled.

SCALA, Space Charge Analysis

Introduction

This section describes SCALA, the electrostatic 3D space charge beam and field analysis program. The program represents the results of many years research, development and application experience.

SCALA can be used to compute electrostatic fields in three dimensions, including the effects caused by space charge in beams of charged particles. The program incorporates state of the art algorithms for the calculation of electromagnetic fields, advanced finite element and non-linear equation numerical analysis procedures. In the following sections the algorithms used in the Space Charge program are described so that users are able to relate the finite element model to their application problem.

The SCALA Algorithm

SCALA uses the finite element method to solve the electrostatic Poisson's equation, and calculate the electric scalar potential (see the description on the use of scalar potentials in section [“TOTAL and REDUCED Magnetic Scalar Potentials”](#) on page 6-9).

The space charge density, included in the Poisson's equation solution, is found by calculating the trajectories of a set of charged particles from the emitters under the influence of the electrostatic field and the magnetic field from any conductors included with the data. (See the **SOLVERS** command in the OPERA-3d Reference Manual for another way of supplying a magnetic field). The particle trajectory calculations include full relativistic correction. The characteristics of the emitter are used to associate a number density (and hence a current and charge density) to the trajectory. The trajectories are intersected with the finite element mesh in order to allocate their space charge to the nodes of the mesh.

Different types of particle emission models can be selected in the analysis program. These include Child's law and Langmuir/Fry relationships for the calculation of the space charge limited current, field emission relationships and defined current densities and initial energy. Several emitting surfaces can be specified in a model, the type of particle and the emission model used can be selected for each surface. The emitting surfaces may overlap each other.

Thermal emission models

Thermal saturation limit - type 0

The thermal saturation limit assumes that the electron current emitted by an electrode is independent of the applied voltage, and that it depends only on the temperature, work function and emission constant of the material. The current density is given by the Richardson-Dushman law

$$j_e = AT^2 e^{\frac{-q_e \phi_w}{kT}} \quad (6.80)$$

where A is the emission constant for the surface in Amps/cm², ϕ_w is the work function of the cathode material in volts, q_e is the electronic charge in coulombs, k is Boltzmann's constant (1.3804×10^{-23} Joule/Kelvin) and T is the temperature of the cathode.

The initial velocity of the particles is assumed to be the mean of the thermal velocity Maxwellian distribution.

Child's Law current limit - type 2

Child's law gives the maximum current density that can be carried in a beam of charged particles across a one dimensional accelerating gap. The equation is derived by requiring equilibrium of the charged particles with a self consistent space charge field.

In order to apply this equation within the program, an accelerating gap width d must be supplied, see Figure 6.1. The equation only applies to infinite planar emitters. It is assumed that the radius of curvature of the emitting surface is large compared to the dimension d and that therefore the one dimensional solution can be used.

The equation for the space charge limited current is

$$j_e = \frac{4\epsilon_0}{9} \sqrt{\frac{2Zq}{m_0}} \frac{V_0^{3/2}}{d^2} \quad (6.81)$$

where ϵ_0 is the permittivity of free space, Zq is the charge on the particle in coulombs, m_0 is the particle rest mass in kilograms and V_0 is the accelerating voltage applied to the accelerating gap d .

When this model of the emitter is used in the space charge solution program, the accelerating voltage is determined by calculating the voltage at a distance d normal to emitter surface, from the finite element solution.

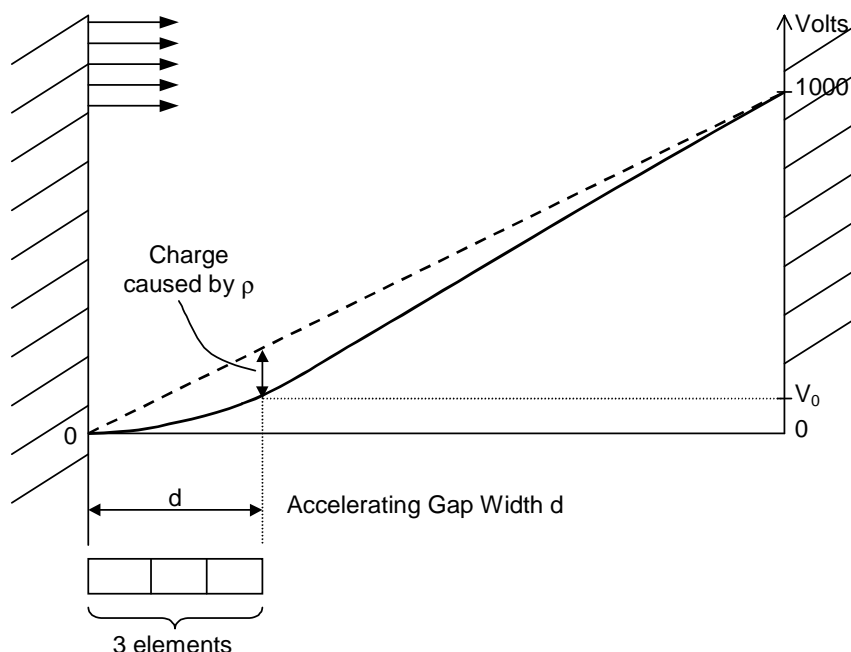


Figure 6.1 Potential distribution between two electrodes for a Child's Law emitter. The emitter on the left-hand side has a potential of 0 volt. The other electrode on the right-hand side has a potential of 1000 volt.

The distance d should be such that two or three elements are included between the sample point and the surface, it should also be small compared to the radius of curvature of the surface.

The initial velocity of the particles is assumed to be the mean of the thermal velocity maxwellian distribution. The emission constant, work function and temperature must be given, and if the Child's law current density exceeds the thermal saturation limit, the current density will be limited to the saturation value.

For emitter types 1, 2 and 8 the current from the emitter is equal to the current actually leaving the emitter. Any test beamlet (particle) that is returned to the emitter by the potential barrier is not included in the total current from the emitter that the program prints out in the *.res* file.

Langmuir/Fry current limit - types 1 and 8

Child's law assumes zero initial energy particles. A more realistic one dimensional solution can be found taking into account the velocity distribution of particles in a thermionic emitter.

In many cases thermionic emitters are operated in a space charge limited mode. This produces a uniform current distribution because the flow is insensitive to

local variations of the surface emissivity. The initial particle energy distribution is important in this case.

A non-linear differential equation must be solved in order to calculate the space charge limited current (Reference: Kirstein, Kino & Waters, Space Charge Flow, McGraw-Hill, pp265-276), the program solves the non-linear equation using a shooting method.

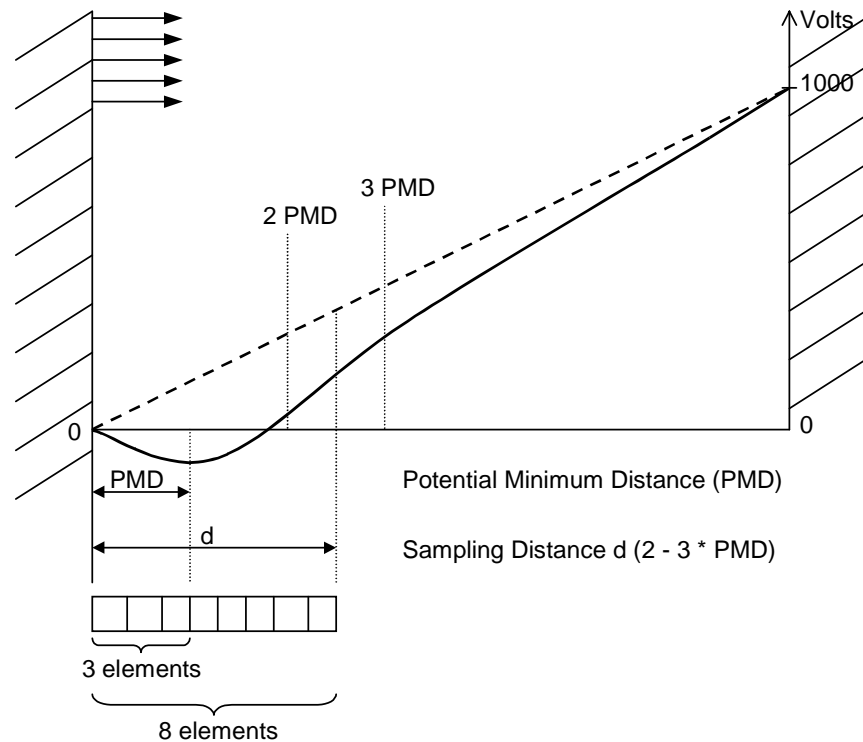


Figure 6.2 Potential distribution between two electrodes. The emitter on the left-hand side has a potential of 0 volt. The other electrode on the right-hand side has a potential of 1000 volt. The space charge in front of the emitter is generating a potential barrier.

As in the Child's law model, a sample distance d must be specified, see Figure 6.2. The sampling distance should be 2 to 3 times the value of the potential minimum distance.

The one dimensional Langmuir/Fry model is then solved using the voltage computed at the sample point.

The initial velocity of the particles is computed from the displaced Maxwellian distribution of the particles that escape the voltage minimum that exists in front of the emitter.

Two options are supported with the Langmuir/Fry emission model. The particles are either tracked from the specified emission surface (type 8) or from a virtual cathode (type 1).

The virtual cathode is defined by a set of positions on the normals from points on the emission surface, where the voltage is equal to the value at the emission surface. This assumes that the space charge is creating a voltage minimum in front of the emitter, if the voltage minimum does not exist then particles are tracked from the emitter surface.

It should be noted that the sample distance d must be larger than the spacing between the emission surface and the computed virtual cathode.

For emitter types 1, 2 and 8 the current from the emitter is equal to the current actually leaving the emitter. Any test beamlet (particle) that is returned to the emitter by the potential barrier is not included in the total current from the emitter that the program prints out in the *.res* file.

Maxwell velocity distribution sampling – type 10 and 11

This option samples the velocity distribution of the electrons escaping from a thermionic emitter. Normal and tangential velocities may be sampled in the current release of the software.

The Maxwellian velocity distribution is divided into a set of ranges (also called bins) such that each range contains the same number of particles. Figure 6.3 shows an example of a velocity distribution with 4 bins. A test ray is tracked from each velocity bin, its velocity is determined by the average kinetic energy of the particles in the bin.

Emitter type 10 uses only a normal velocity sampling; emitter type 11 uses both normal and tangential velocity sampling (the Maxwell distributions for the normal and tangential directions are independent).

The current density of electrons at a particular velocity (v) is $j_e * n(v)$ where

$$j_e = AT^2 e^{\frac{-q_e \phi_w}{kT}} \quad (6.82)$$

and the normalised distribution function in terms of velocity is

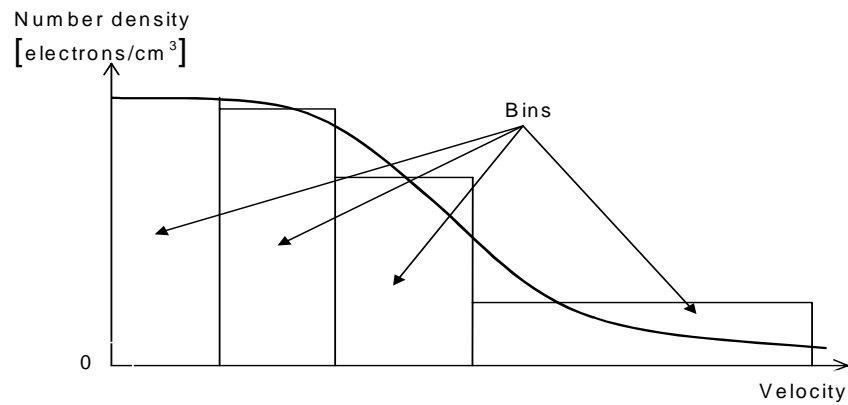


Figure 6.3 Maxwell velocity distribution with 4 ranges

$$n(v) = 2 \sqrt{\frac{m}{2\pi kT}} e^{\frac{-mv^2}{2kT}} \quad (6.83)$$

Depending on the specific geometry of a finite element model some space charge may build up in front of an emitter. In this case the cathode's current is “space charge limited”; there will be a potential barrier in front of the cathode created by the space charge, see Figure 6.4.

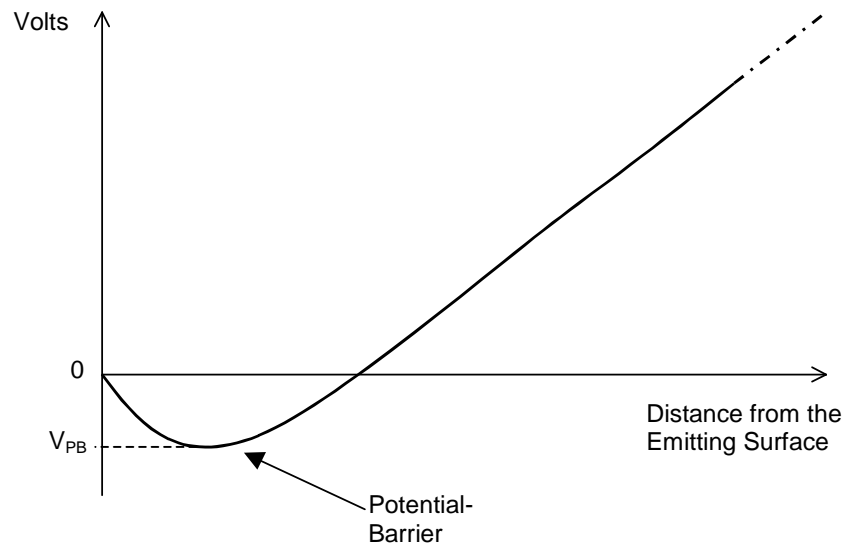


Figure 6.4 Potential barrier in front of an emitter

In order to leave the emitter surface, an electron needs an initial velocity (kinetic energy) greater than that required to pass over the potential barrier.

If the current is space charge limited, the finite element mesh close to the emitter surface must be capable of modelling the voltage minimum that may occur.

With a given value of the potential barrier V_{PB} the escape-velocity can be worked out, see Figure 6.5. In the given example four test trajectories are being emitted per emitting point, but only two of these are capable of passing the potential barrier.

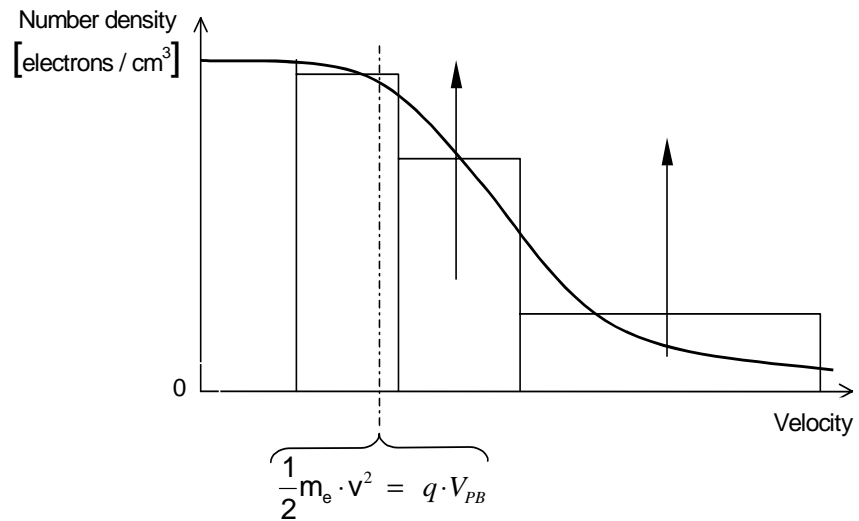


Figure 6.5 Two trajectories with energy greater than V_{PB} are leaving the example-emitter of Figure 6.3

A test ray from a discrete normal velocity bin can escape the potential barrier only if

$$v_{normal} > \frac{2 \cdot q \cdot V_{PB}}{m_e} \quad (6.84)$$

The potential barrier doesn't affect the tangential velocities of emitter type 11. The only fact which determines whether a particle escapes or not is the normal velocity.

Example for emitter type 11

If you ask the program for a normal sampling = 4 and a tangential sampling = 2, the Maxwell velocity distribution will give $4 \cdot 2 \cdot 2 \cdot 4 = 64$ test-rays per emitting point, but only 32 of these can escape the potential barrier in Figure 6.5.

The factors above are based on

- 4 normal samplings for the example in Figure 6.3,
- 2 beams tangential (in x direction),
- 2 beams tangential (in y direction) and
- 4 combinations (x+y, x-y, -x+y, -x-y).

The resolution of the emission current is defined by the equation

$$Resolution = \frac{I_{Saturation}}{number\ of\ v_{normal}\ bins} \quad (6.85)$$

For a full nonlinear solution a sampling of the normal velocity should be expected to require more than 10 bins.

Field effect emission models

Fowler Nordheim Field emission - type 4

This option calculates the current that will be extracted by very high electric fields applied to the surface of the cathode. Significant tunnelling of electrons close to the surface of a cold metallic cathode will occur if a high electric field is applied to materials with a low thermionic work function. Fowler and Nordheim (1928) derived a rigorous solution for this tunnelling current:

$$J_e = 6.2 \times 10^2 \left(\frac{E_f}{\phi_w} \right)^{\frac{1}{2}} \frac{E^2}{E_f + \phi_w} e^{\frac{-6.83 \times 10^7 \phi_w^{\frac{3}{2}}}{E}} \quad (6.86)$$

where J_e is the current density in *Amps/cm²*, E_f is the Fermi energy of electrons in the metal and E is the electric field in *Volts/Metre* applied to the metallic surface. Extensions to the basic theory to include the change in barrier voltage produced by the escaping electrons give the following result:

$$J_e = A \left(\frac{hqE}{4\pi K} \right)^2 \cdot \frac{1}{2qm_0\phi_w} e^{\frac{4\pi b(2qm_0)^{1/2}\phi_w^{3/2}}{Eh}} \quad (6.87)$$

where h is Planck's constant, and:

$$b = \frac{2 \left(1 - \left(\frac{(qE)^{\frac{1}{2}}}{(4\pi\epsilon_0)^{\frac{1}{2}}\phi_w} \right)^{1.69} \right)}{3 \left(1 + 0.1107 \left(\frac{(qE)^{\frac{1}{2}}}{(4\pi\epsilon_0)^{\frac{1}{2}}\phi_w} \right)^{1.33} \right)} \quad (6.88)$$

**Schottky Field
emission - type 5**

This option calculates the current that will be emitted from a cathode, at a known temperature, for the lowest field strengths at which field emission occurs. The current density (J_s in *Amps/cm²*) is given by

$$J_s = AT^2 e^{\frac{-q_e\phi_w + (q_e^3 E)^{\frac{1}{2}}}{KT}} \quad (6.89)$$

**Extended
Schottky Field
emission - type 6**

At intermediate fields, the extended Schottky model predicts higher and more realistic currents. In this case the current density is given by

$$J_e = J_s \frac{\frac{\pi h q_e^{1/4} E^{3/4}}{\pi(2m)^{1/2} KT}}{\sin \frac{\pi h q_e^{1/4} E^{3/4}}{\pi(2m)^{1/2} KT}} \quad (6.90)$$

where J_s is given by equation 6.89

**Automatic Field
Emission
selection - type 7**

Three regimes of field emission are available in the software, the choice depends on the magnitude of the electric field at the start point of each particle. Users may either select a particular model, as shown above, or allow the program to determine which model is appropriate (type 7). The program will select the model that gives the largest current density.

Field Enhancement Factor

The value of electric field, E , computed by SCALA and used in the expressions for the field effect emissions (emitter types 4, 5, 6 and 7) may be multiplied by a user defined field enhancement factor. This allows the user to simply modify the emission model without re-specifying all the emission surface parameters. The default value is 1.0 if not specified. See “Record 3” on page 6-67.

Other emission models

Specified current density - types 3 and 20

The program has options that allows a current density and initial particle energy to be specified.

Plasma Free Surface Model - type 102

A simple model has been introduced that allows extraction of particles from plasmas to be modelled. The emission boundary of the plasma is defined as the surface on which the normal component of the electric field strength is constant. The first point of the user specified emitter surface is kept fixed, the other points are adjusted to achieve this constant field condition. Child's law is then used to compute the current density that can be extracted from each point on the plasma emission boundary.

Emitter Current

Emitter types 1, 2 and 8

The current from the emitter is equal to the current actually leaving the emitter. Any test beamlet (particle) that is returned to the emitter by the potential barrier is not included in the total current from the emitter that the program prints out in the *.res* file.

Emitter types 10 and 11

The total emitted current is printed without subtracting particles that return to the emitter. It is necessary to use the post processor `VIEW INTERSECTION` command to calculate the current in the beam.

Emitter Data Files

The emitter surfaces must be described in a datafile with a filename extension *.emit*, and the same filename as the model geometry file.

The first seven lines of the *.emit* file are referred to as Records 1 to 7. In Record 7, one of five different face segment geometry types can be chosen, which will then be described in detail as Records 8.0 to 8.4. Four of these definitions are defined by coordinates (described in Records 8.0 to 8.3). The emitting surface can also be defined by a labelled face (described in Record 8.4). The label in the *.emit* file must match the label used in the model.

N.B. CGS units must be used in the *.emit* data file.

The *.emit* datafile format is as follows (note that free format input is used and all data items must therefore be entered):

Record 1

Model symmetry

This data specifies how a complete model should be created from the section that has been discretised. Where the model and field exhibit some symmetry it would be costly to discretise the complete space and unnecessary. The discretisation should model the minimal symmetric part of the complete object, but the field must also share the same symmetry. For example, if the geometry and field have a plane of mirror image symmetry, then only half the complete model needs to be specified. The method used to add the space charge from the particle beams into the finite element model also takes advantage of symmetry, only the sections of the emitters inside the discretised space should be described in the *.emit* data file.

Field Number	Type	Units	Description
1	Integer	None	Order of rotational symmetry about the Z axis. If the sign is negative then the voltage changes sign in each section.
2	Integer	None	Reflection flag for XY plane. 0 = no reflection 1 = positive mirror image -1 = inverted mirror image
3	Integer	None	Reflection flag for YZ plane. 0 = no reflection 1 = positive mirror image -1 = inverted mirror image
4	Integer	None	Reflection flag for ZX plane. 0 = no reflection 1 = positive mirror image -1 = inverted mirror image

Record 2**Global parameters**

A number of independent emitting surfaces may be defined in a model. Each surface has its own emission characteristics. Some parameters apply to all aspects of the calculation, for example, the maximum distance between the points used to represent each trajectory, the accuracy of the trajectory calculation and the distance used to evaluate the current limit models.

Field Number	Type	Units	Description
1	Integer	None	Number of Emitters
2	Real	cm	Maximum step length allowed in the trajectory calculation. Note that there is a limit of 5000 steps per trajectory.
3	Real		Absolute tolerance for the trajectory calculation.
4	Real	cm	Normal sampling distance. The distance from the emitter surface used to sample the voltage for the Child's and Langmuir/Fry equations.

Records 3 to 8 are then repeated as a group, Number of Emitter times.

Record 3**Emitter characteristics**

The emission model is specified for each emitter, together with the data required to characterise the emitter.

Field Number	Type	Units	Description
1	Integer	None	Emitter type: 0 = Thermal saturation limit 1 = 1D Langmuir/Fry limit 2 = 1D Child's law limit 3 = specified current density 4 = Fowler Nordheim Field emission 5 = Schottky Field emission 6 = Extended Schottky field emission 7 = Automatic selection between types 4, 5 and 6 8 = 1D Langmuir/Fry limit 10= Maxwell Normal Velocity sampling 11= Maxwell Normal and Tangential Velocity sampling 20= Beamlets with defined position, velocity and current 102= Plasma free surface model
2 ^a	Real	kelvin	Emitter temperature
3 ^a	Real	volt	Emitter Work function (particle energy for emitter type 3)
4 ^a	Real	amp cm ⁻²	Material's Emission constant (A in eqn 6.80). (Current density for type 3 emitter)
5	Real	None	Field enhancement factor for field effect emitters types 4 through 7. If omitted, the default value 1.0 is used.

- a. The value given for Field 2 has no effect with emitter type 3. The values given for Fields 2, 3 and 4 have no effect with emitter type 20. However, values (for example 0.0) must be provided.

Record 4**Particle type**

Field Number	Type	Units	Description
1	Real	None	Particle rest mass in electron rest mass units.
2	Integer	None	Number of charge quanta carried by the particle (-1 for an electron).

Record 5**Sample rays**

Enhanced models for the emitters have been developed. These include sampling of the velocity distribution of the input particles. Trajectories are calculated for a set of particles. In the curved face emitter model at least one particle is started from each subdivision panel of the face. The maximum distance parameter can be used to increase the number used.

Field Number	Type	Units	Description
1	Integer	None	Number of sample bins used to sample tangential velocity (Only used with emitter type 11).
2	Integer	None	Number of sample bins used to sample normal velocity (Only used with emitter types 10 and 11).
3	Real	cm	Maximum distance (tangential to the emitter surface) between sample rays. (Not relevant for emitter type 20)

Record 6**Number of Face segments**

Number of face segments representing the emitter and the local coordinate system that the emitter is defined in.

Each emitter can be modelled as a set of faces. The emitter is defined with respect to a local coordinate system (selected in order to orientate the face correctly).

Field Number	Type	Units	Description
1	Integer	None	The NUMBER of Faces or Beamlets in the model for the emitter.
2	Real	cms	The global X coordinate of the origin of the local system
3	Real	cms	The global Y coordinate of the origin of the local system
4	Real	cms	The global Z coordinate of the origin of the local system
5	Real	degrees	The THETA Euler angle
5	Real	degrees	The PHI Euler angle
5	Real	degrees	The PSI Euler angle

Records 7 and 8 are repeated as a group, Number of Faces/beamlets times.

Record 7

Face segment geometry type

Field Number	Type	Units	Description
1	Integer	None	Geometry type for the face segment 0 = Extruded curved line segment 1 = 4 point ruled quadrilateral facet 2 = 8 point curved quadrilateral facet 3 = A set of points and beamlet data (only valid for emitter type 20) 4 = A labelled face

A ruled surface is generated by a family of straight lines (for a description see, I D Faux and M J Pratt, *Computational Geometry for Design and Manufacture*, published by Ellis Horwood Ltd., 1979, ISBN 0-85312-114-1).

Record 8

Extruded curved line segment definition

A line segment is defined by the X and Y coordinates of its end points in a local coordinate system XY plane. The line is extruded between the two local Z coordinates to create a face. The face is divided into facets as defined by the line and extrusion subdivision parameter. At least one sample particle is tracked from each

facet, unless the facet size is greater than the maximum distance parameter. The subdivision and maximum distance parameters should be such that a reasonable number of sample rays are used, the spacing should be less than the element size.

The bias parameter on the line can be used to produce a non-uniform distribution of particles along the line. If the bias is less than 0.5 the particles will be biased towards the first point.

Field Number	Type	Units	Description
1	Real	cm	Starting X coordinate of the line.
2	Real	cm	Starting Y coordinate of the line.
3	Real	cm	Final X coordinate of the line.
4	Real	cm	Final Y coordinate of the line.
5	Real	1/cm	Curvature of the line (positive implies centre is to the right of the line from start to end).
6	Real	cm	Bias parameter (0.5 for uniform distribution).
7	Integer	None	Line subdivision. Number of segments the line will be divided into (for best results use line segments that correspond with region edges and have the same subdivision).
8	Real	cm	Starting Z coordinate of the extrusion.
9	Real	cm	Final Z coordinate of the extrusion.
10	Integer	None	Number of divisions in the extrusion direction.

Record 8.1

4 point ruled facet definition

The emitting surface may be modelled with an assembly of quadrilateral facets, defined by their four vertex coordinates. The edges of such facets are straight lines, and the surface is ruled. The facets may be degenerated to triangles by defining two consecutive, coincident points.

At least one sample particle is tracked from each sub-facet, unless the sub-facet size is greater than the maximum distance parameter (Record type 5). The maximum distance parameters should be such that a reasonable number of sample rays are used. In general, with this type of emitter geometry, the sub-facets should be coincident with a facet of the finite element model, and the maximum distance parameter should be equal to or less than the finite element size. This will ensure

that all finite elements at the emitter surface are intersected by particles that are used to sample the emission.

Field Number	Type	Units	Description
1	Integer	none	Division into sub-facets, along edges 1 and 3.
2	Integer	none	Division into sub-facets, along edges 2 and 4.

The facet is regularly subdivided into sub-facets, according to the above parameters. Edge 1 joins vertices 1 and 2; edge 2 joins vertices 2 and 3; edge 3 joins vertices 3 and 4; edge 4 joins vertices 4 and 1.

Note that the vertex ordering is important. Particles leave the surface in the direction of its outward normal, defined by a right hand screw convention rotating in the direction of increasing vertex number.

Field Number	Type	Units	Description
1	Real	cm	X coordinate of the first vertex.
2	Real	cm	Y coordinate of the first vertex.
3	Real	cm	Z coordinate of the first vertex.

Field Number	Type	Units	Description
1	Real	cm	X coordinate of the second vertex.
2	Real	cm	Y coordinate of the second vertex.
3	Real	cm	Z coordinate of the second vertex.

Field Number	Type	Units	Description
1	Real	cm	X coordinate of the third vertex.
2	Real	cm	Y coordinate of the third vertex.
3	Real	cm	Z coordinate of the third vertex.

Field Number	Type	Units	Description
1	Real	cm	X coordinate of the fourth vertex.
2	Real	cm	Y coordinate of the fourth vertex.
3	Real	cm	Z coordinate of the fourth vertex.

Record 8.2**8 point ruled facet definition**

The emitting surface may be modelled with an assembly of quadrilateral facets, defined by their four vertex coordinates and four midside coordinates. The edges of such facets are curved, and the surface is curved. The shape of the facet is isoparametric, using a tri-quadratic mapping, equivalent to the surface shape obtained with a 20 node isoparametric brick finite element. The facets may be degenerated to triangles by defining two consecutive, coincident vertex points, that are also coincident with the midside point on the edge between them.

At least one sample particle is tracked from each sub-facet, unless the sub-facet size is greater than the maximum distance parameter (Record type 5). The maximum distance parameters should be such that a reasonable number of sample rays are used. In general, with this type of emitter, the sub-facets should be coincident with facets of the finite element model, and the maximum distance parameter should be equal to or less than the finite element size. This will ensure that all finite elements at the emitter surface are intersected by particles that are used to sample the emission.

Field Number	Type	Units	Description
1	Integer	none	Division into sub-facets, along edges 1 and 3.
2	Integer	none	Division into sub-facets, along edges 2 and 4.

The facet is regularly subdivided into sub-facets, according to the above parameters. Edge 1 joins vertices 1 and 2; edge 2 joins vertices 2 and 3; edge 3 joins vertices 3 and 4; edge 4 joins vertices 4 and 1.

Note that the vertex ordering is important. Particles leave the surface in the direction of its outward normal, defined by a right hand screw convention rotating in the direction of increasing vertex number.

Field Number	Type	Units	Description
1	Real	cm	X coordinate of the first vertex.
2	Real	cm	Y coordinate of the first vertex.
3	Real	cm	Z coordinate of the first vertex.

Field Number	Type	Units	Description
1	Real	cm	X coordinate of the second vertex.
2	Real	cm	Y coordinate of the second vertex.
3	Real	cm	Z coordinate of the second vertex.

Field Number	Type	Units	Description
1	Real	cm	X coordinate of the third vertex.
2	Real	cm	Y coordinate of the third vertex.
3	Real	cm	Z coordinate of the third vertex.

Field Number	Type	Units	Description
1	Real	cm	X coordinate of the fourth vertex.
2	Real	cm	Y coordinate of the fourth vertex.
3	Real	cm	Z coordinate of the fourth vertex.

Field Number	Type	Units	Description
1	Real	cm	X coordinate of midside node between vertices 1 and 2.
2	Real	cm	Y coordinate of midside node between vertices 1 and 2.
3	Real	cm	Z coordinate of midside node between vertices 1 and 2.

Field Number	Type	Units	Description
1	Real	cm	X coordinate of midside node between vertices 2 and 3.
2	Real	cm	Y coordinate of midside node between vertices 2 and 3.
3	Real	cm	Z coordinate of midside node between vertices 2 and 3.

Field Number	Type	Units	Description
1	Real	cm	X coordinate of midside node between vertices 3 and 4.
2	Real	cm	Y coordinate of midside node between vertices 3 and 4.
3	Real	cm	Z coordinate of midside node between vertices 3 and 4.

Field Number	Type	Units	Description
1	Real	cm	X coordinate of midside node between vertices 4 and 1.
2	Real	cm	Y coordinate of midside node between vertices 4 and 1.
3	Real	cm	Z coordinate of midside node between vertices 4 and 1.

Record 8.3

Beamlet with defined direction and current

This option can only be used with emitter type 20.

The data describes each beamlet that will be included in the calculation. The starting position, velocities and current in the beamlets all must be specified. Note that the beamlets are specified in the local coordinate system defined in Record 6, this affects both the coordinates and the velocities.

Field Number	Type	Units	Description
1	Integer	Amps	The current in the beamlet
2	Integer	cm	The local X coordinate of the start of the beamlet.
3	Integer	cm	The local Y coordinate of the start of the beamlet.
4	Integer	cm	The local Z coordinate of the start of the beamlet.
5	Integer	cm/sec	Initial local X velocity of the particles in the beamlet.
6	Integer	cm/sec	Initial local Y velocity of the particles in the beamlet.
7	Integer	cm/sec	Initial local Z velocity of the particles in the beamlet.

Record 8.4

Labelled Face

Additional labels may be added to a surface in the Modeller or pre processor to allow the labelled surface(s) to be used as emitters. SCALA will ensure that at least one trajectory leaves every element on the surface, although this will be superseded by the maximum distance between sample rays specified in Record 6

(see “[Sample rays](#)” on page 6-68), if this is smaller than the element size. Particles will be emitted from the surface in the outward normal direction of the surface if only the surface label is given or inward normal if the label is preceded by a minus (-). “Toggling” of the outward normal direction of surfaces in the modeller to ensure that all surfaces with the label are emitting in the same direction may be necessary.

Field Number	Type	Units	Description
1	Character	none	Additional label on surface (Note: This type of emitter cannot be used for emitter type 20).

Example data files

An example *.emit* datafile defining a plane emitting surface (in an XY plane of the global coordinate system) which emits particles in the positive Z direction of the global coordinate system. The emitting surface is modelled using an extruded line.

```

1      0      1      0
1      0.2    0.001  0.004
1      1250.0 1.86   350.0
1.0    -1.0
1      1      0.05
1      0.0    5.15   0.001  90   -90   90
0
0.0    0.0    0.085  0.0    0.0    0.5   6   -0.5  0.5   8

```

An example *.emit* datafile defining an eight node curved quadrilateral facet as an emitter, the facet is triangular with points 1, 4 and 8 coincident.

```

4      0      0      1
1      0.005   0.011  0.001E-5
4      300.0   4.52   120.0
1.0    -1.0
1      1      0.4E-6
1      0.0    0.0      0.0      0.0  0.0  0.0
2      (Record 7: 8 node facet geometry)
30     4      (Record 8.1.0: sub facet division)
0.0    0.0    5.01E-6

```

4	0	0	1
2.505E-6	0.0	4.339E-6	
1.7713E-6	1.7713E-6	4.339E-6	
0.0	0.0	5.01E-6	
1.297E-6	0.0	4.8393E-6	
2.3144E-6	0.9586E-6	4.339E-6	
0.9171E-6	0.9171E-6	4.8393E-6	
0.0	0.0	5.01E-6	

An example *.emit* datafile defining a set of beamlets with specified current, position and velocity.

4	0	0	1			
1	0.005	0.011	0.001E-5			
20	0.0	0.0	0.0	(Emitter type 20 - Fields 2 to 4 not used)		
1.0	-1.0					
1	1	0.0	(With Emitter type 20 the maximum spacing is not used)			
4	0.0	0.0	0.0	0.0	0.0	0.0
3	(Record 7: Beamlet data selection)					
1.0E-6	-0.1	-0.1	0.0	0.0	0.0	1.0E+6(Record 8.3:)
3						
1.0E-6	-0.1	0.1	0.0	0.0	0.0	1.0E+6
3						
1.0E-6	0.1	0.1	0.0	0.0	0.0	1.0E+6
3						
1.0E-6	0.1	-0.1	0.0	0.0	0.0	1.0E+6

An example *.emit* datafile defining a labelled surface.

```

0      0      1      0
1      0.05    0.05    0.01
1      2600.0  4.25    120.0
1.0    -1.0
1      1      0.1      (With a labelled surface emitter the maximum
                        spacing will only be used if the surface ele-
                        ments are larger than the spacing)
1      0.0      0.0      0.0      0.0      0.0      0.0
4      (Record 7: Labelled face)
LFEMIT (Record 8.4: Surface label with particles emitted in the
        direction of the outward normal

```


Chapter 7

Application Notes

Material Anisotropy to Represent Laminations

In earlier versions of Vector Fields software, an option for modelling laminated materials was available explicitly. In most cases this is still available for compatibility. However the trend is to model laminated materials using the more general anisotropic material definitions. The following describes a method of using the anisotropy feature to model lamination stacks in bulk.

This requires the modification of the BH characteristic by a packing factor (f) from the BH characteristics of the solid material. The modification required depends on the direction of lamination.

In the plane of the lamination

The component of the magnetic field strength tangential to the plane of the lamination, H_t , is continuous at the air/steel (insulation) interface, i.e.

$$\mathbf{H}_t = \frac{\mathbf{B}_{air}}{\mu_0} = \frac{\mathbf{B}_{steel}}{\mu_{steel}} \quad (7.1)$$

such that:

$$\mathbf{B}_{steel} = \frac{\mu_{steel}}{\mu_0} \mathbf{B}_{air} \quad (7.2)$$

The average flux density in the tangential direction, \mathbf{B}_t , is given by:

$$\mathbf{B}_t = f\mathbf{B}_{steel} + (1-f)\mathbf{B}_{air} = \left[f\frac{\mu_{steel}}{\mu_0} + (1-f) \right] \mathbf{B}_{air} \quad (7.3)$$

The effective permeability in the plane of the lamination (tangential to the lamination) is given by:

$$\mu_t = \frac{\mathbf{B}_t}{\mathbf{H}_t} = \frac{\left[f\frac{\mu_{steel}}{\mu_0} + (1-f) \right] \mathbf{B}_{air}}{\frac{\mathbf{B}_{air}}{\mu_0}} = f\mu_{steel} + (1-f)\mu_0 \quad (7.4)$$

In the direction normal to the lamination

The component of flux density normal to the plane of the lamination, \mathbf{B}_n , is continuous at the steel/ air (insulation) interface, i.e.

$$\mathbf{B}_n = \mu_0 \mathbf{H}_{air} = \mu_{steel} \mathbf{H}_{steel} \quad (7.5)$$

Hence the average field strength in the normal direction to the lamination is given by:

$$\mathbf{H}_n = f\mathbf{H}_{steel} + (1-f)\mathbf{H}_{air} = \left[f\frac{\mu_0}{\mu_{steel}} + (1-f) \right] \mathbf{H}_{air} \quad (7.6)$$

Hence the effective permeability in the normal direction is given by:

$$\mu_n = \frac{\mathbf{B}_n}{\mathbf{H}_n} = \frac{\mu_0 \mathbf{H}_{air}}{\left[f\frac{\mu_0}{\mu_{steel}} + (1-f) \right] \mathbf{H}_{air}} = \frac{\mu_{steel} \mu_0}{f\mu_0 + (1-f)\mu_{steel}} \quad (7.7)$$

Combined magnetic and electric fields from TOSCA

The following describes the procedure required to calculate tracks in combined magnetic and electric fields in the OPERA post processor.

1. Solve the magnetostatic and electrostatic problems separately, *mag.op3* and *elec.op3* will be used as the respective file names.
2. Now form the table of electrostatic model coordinates by activating the *elec.op3* file;

Tables → Read and Write table files

Create a Table

Read and write table files

Input data ...

☐ Table file ☐ Internal buffer ☒ Database ☐ Selected surface

Input table file

File name

Output data ...

☒ Table file ☐ Internal buffer ☐ Database ☐ No output

Output table file

File name coords.table

Field components:

1	X	2	Y	3	Z
4		5		6	
7		8		9	
10		11		12	

OK Cancel

The filename *coords.table* is entered only as an example (any filename is valid) and the output components must be **X**, **Y**, **Z**. Click on **OK** to process the Table.

- During the next phase, the unit of flux density must be set to Gauss. Activate the *mag.op3* file and determine the fields at the coords by selecting

Tables → Read and Write table files

Create a Table

Read and write table files

Input data ...

☒ Table file ☐ Internal buffer ☐ Database ☐ Selected surface

Input table file

File name: coords.table

Output data ...

☒ Table file ☐ Internal buffer ☐ Database ☐ No output

Output table file

File name: fields.table

Field components:

1 X	2 Y	3 Z
4 RBX	5 RBY	6 RBZ
7	8	9
10	11	12

OK Cancel

The filename *fields.table* is specified (any filename is valid) and the output components set to **X, Y, Z, RBX, RBY, RBZ**. Click on **OK** to process the Table.

4. Activate the *elec.op3* file and set the magnetic fields at the coords by

Tables → Read and Write table files	
Read a Table	

The combined magnetic and electric fields will be available and may be switched on, when required, in the trajectory menu.

NB: The SCALA analysis program checks for nodal values of **RHX**, **RHY** and **RHZ**, and uses these for overlaid magnetic fields. Values can be placed into the database as described above - see the Reference Manual **TABLE** command for details.

Parameterized Models in OPERA-3d

Geometric Modeller

Command scripts can be written so that the model geometry is defined using parameters. In order to help users achieve this, the Modeller provides a log file containing all the input commands used to create a model. Having once created the model interactively, the log file can be edited to add parameters and thus create a script file that can be used to create a new model.

Pre Processor

1. Create a base “standard” model with no parameterisation in the normal way. Write out the file as a pre processor file, say *base.oppre*.
2. Clear and reset the pre processor. Read the *base.oppre* model into the pre processor.
3. Select **MODIFY**
 - Select **Point coordinates**
 - Set the *plane number* to the plane in which the parameterised coordinates are required.
 - Select **Select/de-select point** and pick the points you wish to move in this plane.
 - Select **Transform points** and choose a method of moving the points.
 - As an example, select **Displace** and enter three values (say) 1.0, 0, 0.
4. Close the menus and write out the new file, say *param.oppre*.
5. Using a system editor to edit the *.oppre* file.
 - At the top of the file (in the first line above the first command) the user defined constant for the displacement `#xdis` should be set (say to 2.54), so enter¹

```
$para #xdis 2.54
```
 - Move to the bottom of the file and then move back up to the line (where “n” is the plane you have chosen²)

-
1. Multiple parameters may be defined in the same file. Using the **\$ASK** command (rather than the **\$CONS** command) will make the *.oppre* file prompt the user for a value of the user constant each time the file is read.

MODIFY STAR=n | POINT

- Move down to the **DISP** command and edit the line to include the parameter for the x displacement:

DISP #xdis 0 0¹

- Save the *.oppre* file. By editing the #xdis value you may create any offset required each time the *.oppre* file is read.

-
2. Using a do-loop in the *.oppre* file will allow the parameterised values to be used in many planes automatically.
 1. Note that there is also a **DISPLAY** command, as well as the **DISPLACE** sub-command in the **MODIFY** command. As commands are truncated to 4 characters, this can lead to confusion searching for the correct line in the *.oppre* file. However, if the model was generated using the GUI, a line with the **DISPLAY** command should read: **DISP SIZE=...**, whereas a line with the **DISPLACE** sub-command should read: **DISP 1 0 0 ...**

Q and g (R/Q) factor from SOPRANO solution

The following describes the methods used to calculate Q and Q/R values for a resonant cavity solution from SOPRANO.

Definitions

Definitions of geometric impedance, g (sometimes known as R/Q) characterizes the efficiency of the geometry for particle acceleration. The units are SI:

$$g = \frac{R}{Q} = \frac{|V|^2}{PQ} \quad (7.8)$$

where

P is the power dissipated in the metallic surfaces

Q is the quality coefficient

V is the accelerating voltage seen by the particles, assumed to cross the whole cell at the speed βc and

$$V = \int E_z \cos \frac{j\omega z}{\beta c} dz \quad (7.9)$$

c is the speed of light (m/s)

β is the ratio of the speed of the particle to the speed of light

ω is the angular resonant frequency

E_z is the z-component of the electric field on the axis.

The Q factor may be calculated as:

$$Q = 2\omega \frac{W}{P} \quad (7.10)$$

where W is the energy in the cavity.

The geometric impedance g is associated with the shunt impedance per unit length, R_l as

$$R_l = \frac{gQ}{L} \quad (7.11)$$

where

the geometric impedance, g, is:

$$g = \frac{R}{Q} = \frac{\left\{ \int_0^{\frac{\lambda}{4}} E_z \cos\left(\frac{\omega z}{\beta c}\right) dz \right\}^2}{PQ} \quad (7.12)$$

L is the cavity length.

P is calculated over all the cavity wall in $\lambda/4$ period.

Command files to compute Q-factor and Geometric Impedance

The following *.comi* file calculates the Q factor of a SOPRANO resonant cavity:

```
/Set SI units
unit len=metr,flux=tesl,fiel=am,elec=vm,
      s=amp,v=wbm,co=sm,cu=am2,p=watt,
      fo=newt,ener=joul

/ Select the PEC (metallic) surfaces

sele type=surf label=pec

/Define variables for |Hxn|**2, omega,
/conductivity (for copper say 5e7 S/m), etc.

$para #hxn (nz*hy-ny*hz)**2+(nx*hz-nz*hx)**2+(ny*hx-nx*hy)**2
$para #omeg 2*pi*freq
$cons #cond 5e7
$para #fact sqrt(#omeg*mu0/2/#cond)
$para #loss #fact*#hxn

/Calculate surface integral and save in variable #i

surf -tave #loss
$cons #i integral

/Calculate magnetic energy

energy

/Calculate Q-factor

$cons #q #omeg*energy*2/#i

/As a check, calculate the electric energy and Q-factor

volu -tave comp=epsilon0/2*emod**2
$cons #q2 #omeg*integral*2/#i
```

The following is additional to the above *.comi* file and calculates the geometric impedance following the above commands:

```
/Calculate line integral for quarter wavelength length
/cavity setting #zend occurs at the peak field
/say 0.15 m) and beta (say 0.7)

$cons #zend 0.15
$cons #beta 0.7
line 0 0 0 0 0 #zend n=100
colo set code=9 1 1 1
plot comp=e*z*cos((z-#zend)*#omeg/#beta/c)

/Calculate the geometric impedance (R/Q),
/using #symm (the symmetry copies around the
/z axis say 8).

$cons #symm 8
$cons #imp integral**2/#i/#symm/2/#q
```

External Magnetic Fields in TOSCA and SCALA

Introduction

It is sometimes convenient to introduce a “background” magnetic field into an analysis. OPERA-3d supports this facility in magnetostatic TOSCA analyses and in SCALA. A uniform field with any orientation may be specified. There are many applications that can use these facilities, but the most common are:

- Including the Earth’s field
- Shielding calculations
- Combined electric and magnetic field particle tracking

External magnetic fields in TOSCA

In TOSCA, the applied magnetic field contributes to the overall coil fields in the problem. It is equivalent to having an infinite solenoid whose axis is oriented in the direction of the field. Consequently, it is necessary for the exterior of the model space (the outer layers of finite elements in the mesh) to use reduced magnetic scalar potential. The extent of this layer is unimportant – a single layer of elements works just as well as making all the air reduced potential. Figures 7.1 and 7.2 show two example models of a block of steel in a uniform Y-directed field for which the results are identical, showing the extent of the reduced potential in each model.

The magnitude and direction of the applied magnetic field is defined by its global X, Y and Z components. The components are specified during the creation of the *.op3* database under the

FILE ↓

...create new database → Analysis data → External fields

menu or the creation of a new case in an existing database using the

FILE ↓

...use existing database → Analysis data → External fields

menu. The values should be specified in Oersted for CGS models or Amps/metre for all other unit systems. Figure 7.3 shows the resulting field for the block models shown in Figures 7.1 and 7.2.

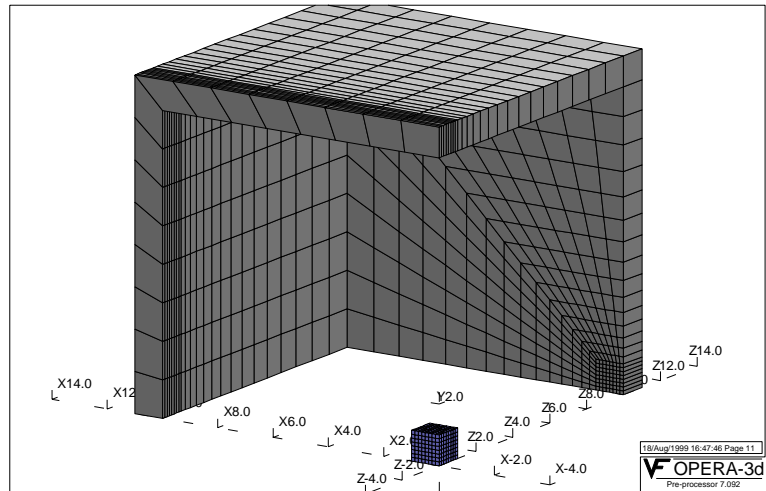


Figure 7.1 Outer layer reduced

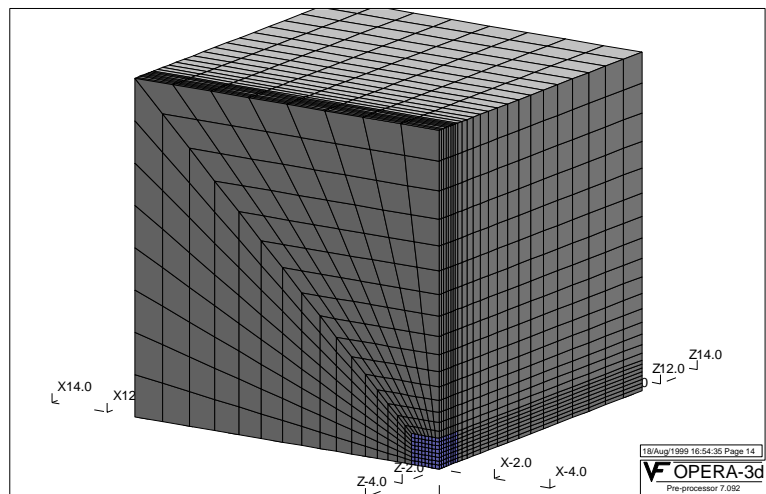


Figure 7.2 All air reduced

External Fields in SCALA

The use of external fields in SCALA is to add a uniform magnetic field to the computed electric field when the particle trajectories are being calculated. In many space charge applications, the dimensions are sufficiently small that any applied magnetic field existing in the same space can be considered uniform over the volume. The applied magnetic field plays no part in the equations solved by

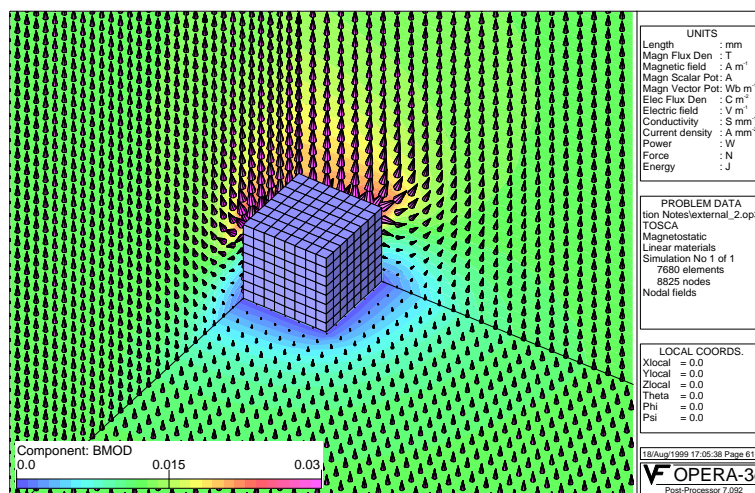


Figure 7.3 Application of external Y-directed field to block

the finite element method but influences the trajectories taken by the particles. Equally, the current in the beam produces zero perturbation on the magnetic field. The external field is added to the *.op3* file for the SCALA analysis through the **FILE** menu, similarly to magnetostatic TOSCA analyses (see above).

The user should take care when adding magnetic fields to a SCALA model. The symmetry that exists in the electric field when the magnetic fields are not present may be disturbed by the asymmetric space charge resulting from the beam. Figure 7.4 shows two views of the emission current density from a simple electron gun. On the right, there is no external magnetic field acting, while on the left a Y-field of -2000 A/m has been added. As can be seen, the 8-fold symmetry obtained in the zero field case is reduced to symmetry in the ZX plane when the magnetic field is applied.

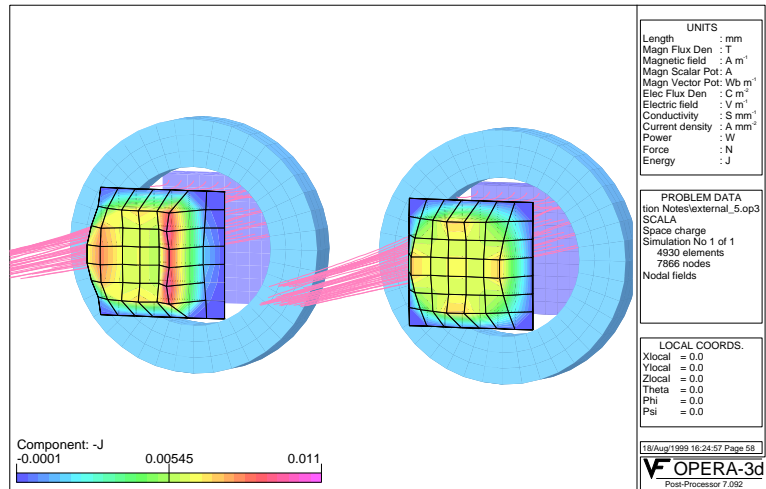


Figure 7.4 Effect of external magnetic field on emitted beam

Multiply Connected Regions

Introduction

The TOSCA, ELEKTRA and CARMEN programs all use magnetic scalar potentials. CARMEN and ELEKTRA also use magnetic vector potentials and electric scalar potentials for conducting regions where unknown (i.e. not source) current distributions exist. The benefit of using magnetic scalar potentials wherever possible is that only a single unknown is solved at each node.

In the conducting regions of ELEKTRA and CARMEN, 4 unknowns (3 components of magnetic vector potential and the electric scalar potential) are needed. However, using magnetic scalar potentials can give rise to problems called “multiply connected regions”. This term means that a magnetic scalar potential domain surrounds a current carrying conductor in such a way that the magnetic scalar potential becomes multi-valued at some point in the domain.

This arises from Ampere’s law, which states that the line integral of the magnetic field strength, \mathbf{H} , around a closed contour is equal to the current enclosed, I , i.e.

$$\oint_C (\mathbf{H} \cdot d\mathbf{l}) = I \quad (7.13)$$

In a total magnetic scalar potential volume, the magnetic field strength is given by the gradient of the total scalar potential, ψ , i.e.

$$\mathbf{H} = -\nabla\psi \quad (7.14)$$

Hence, the contour integral of the gradient of the potential is equal to the current enclosed and this leads to the potential becoming multi-valued at some point.

In practice, however, the potential values in TOSCA, ELEKTRA and CARMEN cannot be multi-valued – only one value is allowed at each node. Consequently, multiply connected solutions show a sudden jump in potential across one element and non-physical fields. A simple example is shown in Figure 7.5 and 7.6.

Automatic Scalar Potential Cuts in TOSCA

In-built pre processor and Modeller routines will correct a TOSCA model that contains multiply connected regions. The software has the capability of checking for multi-valued total magnetic scalar potential regions, and introducing potential

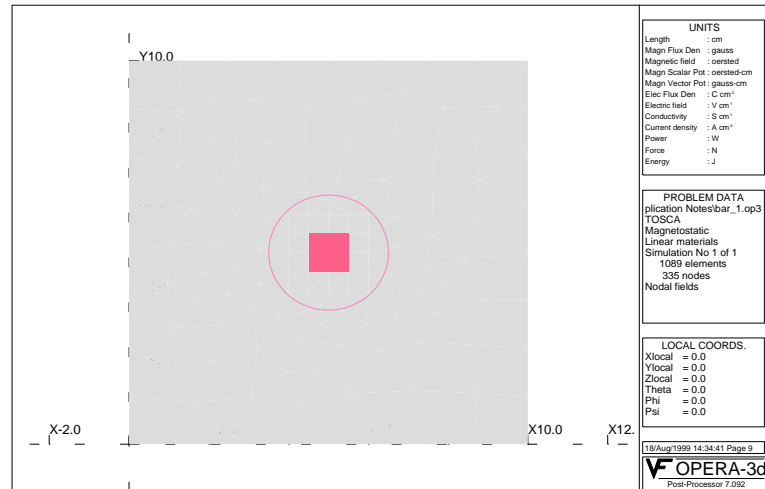


Figure 7.5 Closed circular contour around multiply connected straight conductor

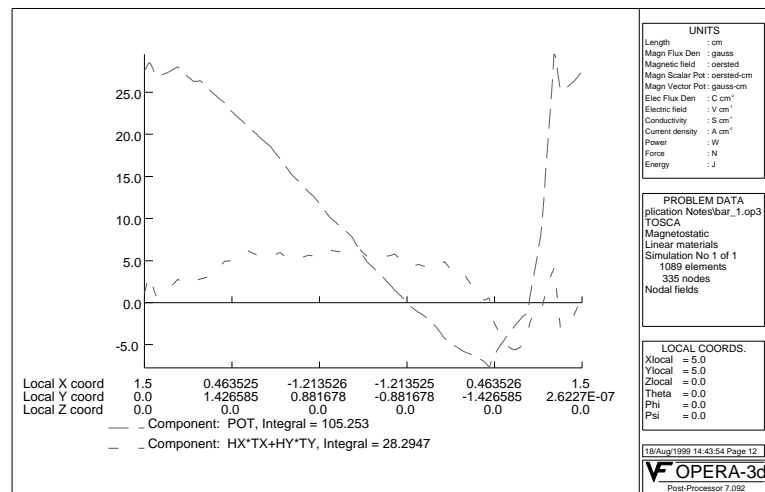


Figure 7.6 Potential and tangential magnetic field on circular path

“cuts” where appropriate in order to rectify the problem. The routines are activated during the database definition stage, although the user does have the option of switching off the “automatic cuts” at a later stage.

The ELEKTRA solver is completely unaffected by these functions. It is therefore still necessary for the user to produce non-multiply connected models.

Source conductors

Source conductors are included in ELEKTRA and CARMEN by ensuring that they are placed within a volume where the reduced magnetic scalar potential, ϕ , is solved. In these volumes the field is partitioned into the field created by the source conductors, \mathbf{H}_s , and the field created by the magnetisation, \mathbf{H}_m , (+ the field created by the unknown currents in ELEKTRA and CARMEN, \mathbf{H}_e) i.e.

$$\mathbf{H} = \mathbf{H}_s + \mathbf{H}_m (+ \mathbf{H}_e) \quad (7.15)$$

The source field is calculated by the Biot-Savart expression and the reduced magnetic scalar potential represents the remainder of the field

$$\mathbf{H} = \mathbf{H}_s - \nabla \phi \quad (7.16)$$

To ensure that a total magnetic scalar potential region is not multiply connected, the reduced potential region should cut through an encircling total potential region to give a total current of zero in the reduced potential region.

A simple magnet example correctly constructed is shown in Figure 7.7, where all of the problem space is total potential apart from the reduced potential region indicated.

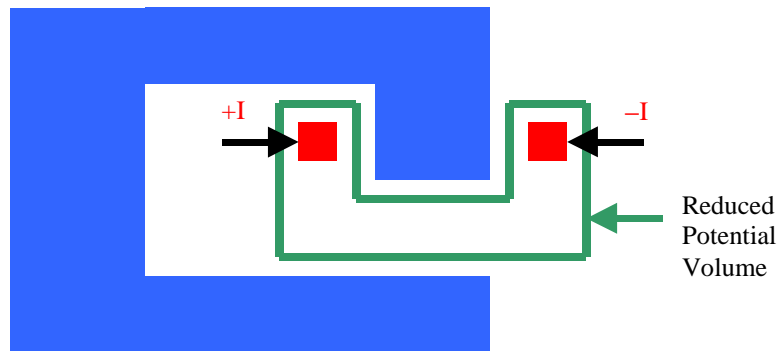


Figure 7.7 Correct use of reduced potentials

Multiply Connected TOTAL Regions

Things are not always as straightforward as in Figure 7.7, where a reduced potential cut can be made within air in order to avoid a multiply connected total potential. A transformer for example has a closed iron yoke (normally a total potential region) surrounded in part by a primary coil carrying a non-zero current (Figure 7.8). Only one quarter of the yoke of the transformer is modelled, using the symmetry of the problem.

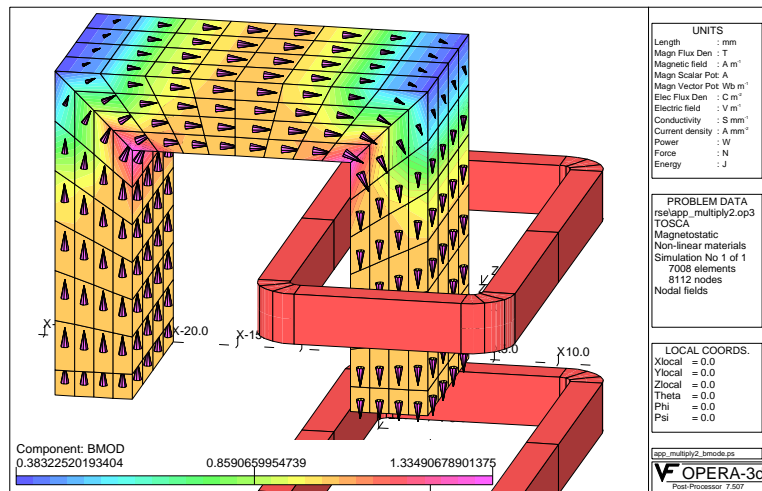


Figure 7.8 Correct flux distribution in the yoke of a transformer

Figure 7.8 also shows the correct flux distribution in the yoke. There is a high flux density near the internal corners of the yoke, and a lower flux density near the outer corners. The model has been set up using a reduced potential cut through the yoke itself, through the limb furthest from the primary coils. The cut volume is defined as **IRON REDUCED**, and the rest of the yoke is **IRON TOTAL**.

Figure 7.9 shows the distribution of the total scalar potential in the yoke. The potential is zero on both sides of the yoke touching the xz-plane (implied by the **NORMAL MAGNETIC** boundary condition). In the cut on the left hand side, the potential drops from its maximum value to zero.

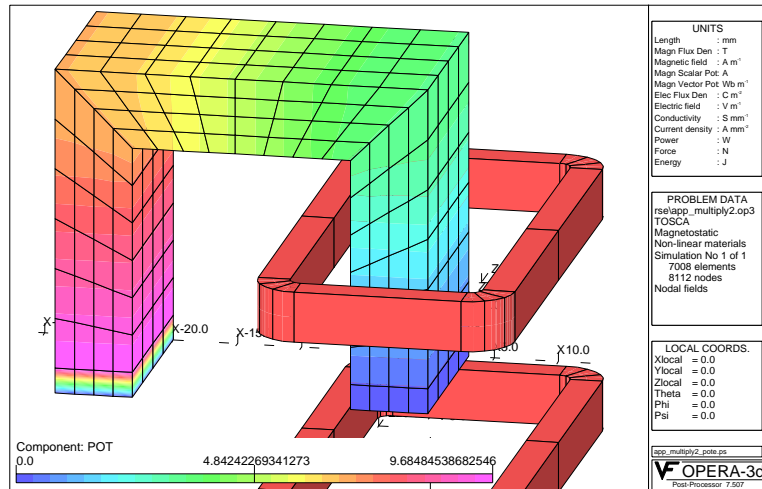


Figure 7.9 Total scalar potential in the yoke with correct cut in place

Incorrect Solution

In order to see how the solution is incorrect without a suitable cut, Figure 7.10 shows the flux density distribution in an incorrectly defined model. In this case there is no reduced potential cut at all. It is only under the coils that the flux is pointing in the correct direction. Everywhere else the flux magnitude and direction is incorrect.

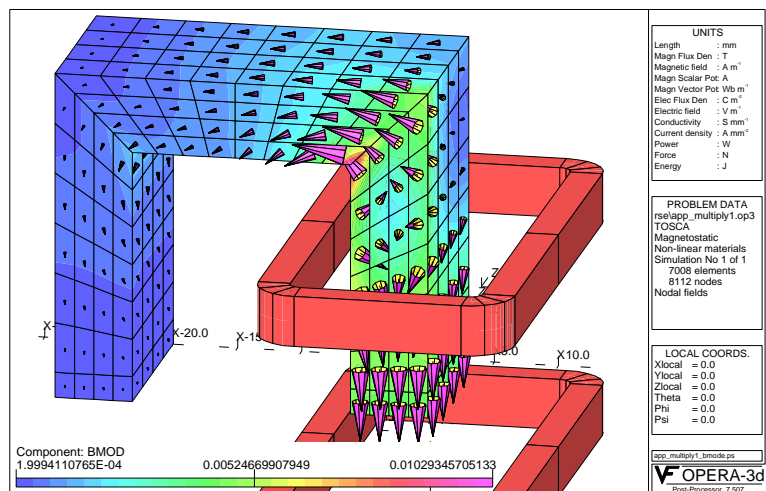


Figure 7.10 Incorrect flux distribution in a model without a cut

The distribution of the total scalar potential is shown in Figure 7.11. It can be seen that the potential on the xz-plane is being forced to zero. This is due to the boundary condition of this plane, which is set to **NORMAL MAGNETIC**.

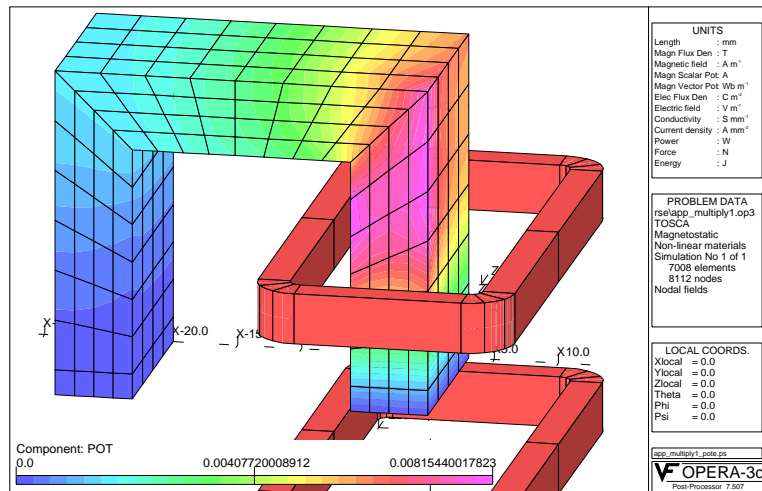


Figure 7.11 Total scalar potential in a model without a cut

Implied Cut in Total Potential Volume

In a problem that has a **NORMAL MAGNETIC** symmetry plane, as with the transformer example above (Figure 7.8), the boundary conditions on the total magnetic potential can be used to imply the cut by the reduced potential region. Consider the yoke and coil in Figure 7.12, which are symmetric about the plane $Y = 0$. The coil, of course, must be reflected to obtain the correct answer.

The **NORMAL MAGNETIC** boundary condition implies the condition on the reduced or total scalar potentials $\phi = 0$ or $\psi = 0$ respectively. However, the total scalar potential must satisfy Ampere's law (see above) which is achieved by replacing the $\psi = 0$ by $\psi = +I$. The implication is that the image in negative Y has the condition $\psi = -I$ at the same boundary i.e. the potential is discontinuous due to an infinitely thin reduced potential cut.

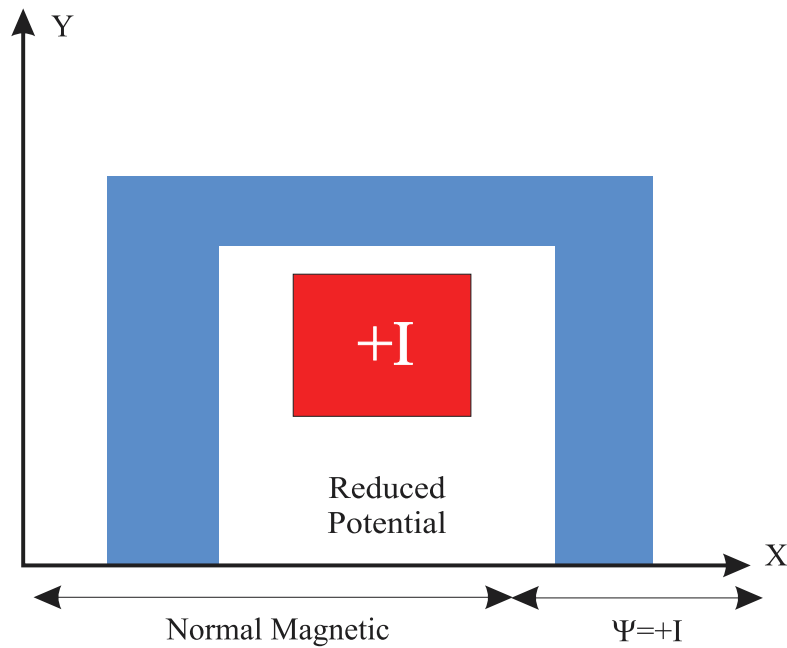


Figure 7.12 Creating a potential jump to ensure Ampere's Law is satisfied

Example

Figure 7.13 shows an example of the use of an implied cut. The value of the total scalar potential on the left-hand side of the yoke has been set to 10, the value on the right-hand side is zero (implied by the NORMAL MAGNETIC boundary condition). A similar result would be obtained if the potentials were set to +5 and -5 Volts respectively - it is the potential difference that is important.

Implied Cut in Vector Potential Volume

If the same problem is to be run in ELEKTRA with the yoke specified as **VECTOR** potential and the surrounding air space both inside and outside the yoke as **REDUCED** potential, it would appear that multi-valuing will not be an issue. In fact, it is important to note that on an interface between reduced and vector potential volumes, ELEKTRA and CARMEN insert an infinitely thin region of total potential. If the user only specifies **NORMAL MAGNETIC** boundary conditions on the $Y = 0$ plane, the boundary condition applied to the total potential at either end of the interface will be $\psi = 0$, which will lead to a jump in potential somewhere on the interface and non-physical results. This can be overcome by inserting a real region of total potential and applying appropriate boundary conditions, as shown in Figure 7.14.

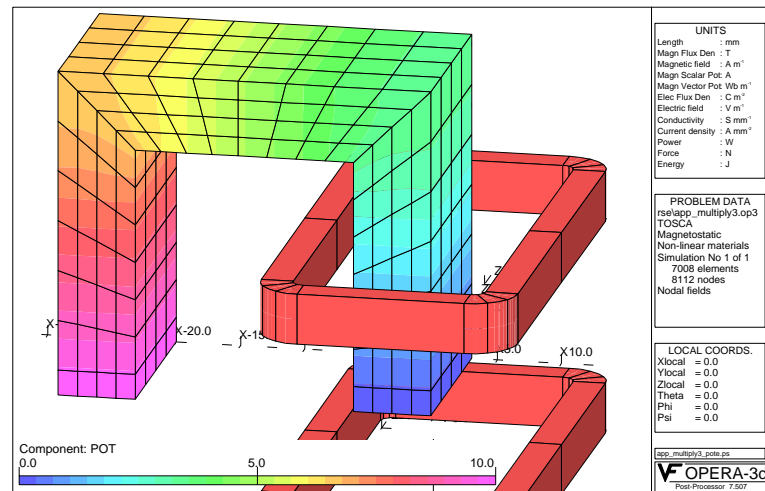


Figure 7.13 Total scalar potential in the yoke with an implied cut

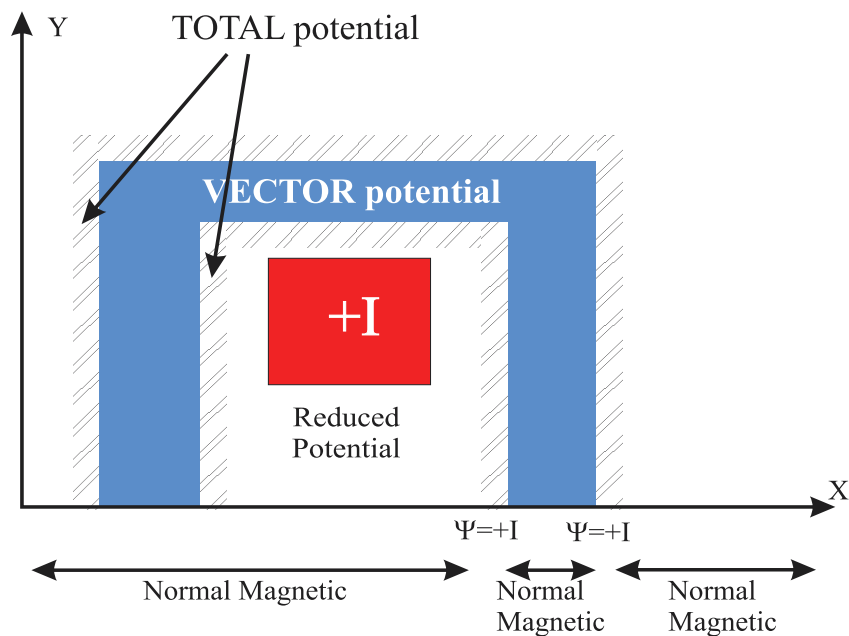


Figure 7.14 Using thin **TOTAL** potential layer to ensure correct modelling when using **VECTOR** potential in material

In ELEKTRA-TR and CARMEN, it is necessary that the value of the boundary condition follows the drive function of the coil. A label should be added to the boundary conditions on the total potential so that the same drive function as the coil can be specified when creating the .op3 file. In ELEKTRA-SS, it is only necessary to use a label on the boundary condition if the phase angle of the current in

the coil is non-zero since the boundary condition must be in-phase with the drive. For ELEKTRA-VL, it is not necessary to specify a label, as all the drives are dc.

Multiply Connected VECTOR Regions

In ELEKTRA and CARMEN regions solved with magnetic vector and electric scalar potentials (**VECTOR** regions) allow currents to flow. These conducting regions may be carrying induced (eddy) currents or a current distribution produced by **VOLTAGE** boundary conditions. It is very common for these regions to consist of a circuit of solid members with non-conducting spaces between them, where the current flows along one member, through a connecting bar and back along another member.

A good example of this is the “cage” of an induction motor, which consists of many conducting bars parallel to the axis of the motor and joined together at their ends by conducting rings. This cage is situated within a laminated iron rotor body, which can be considered as non-conducting since the laminations prevent axial flow of currents.

The user must take care not to completely surround an individual member of one of these circuits with a multiply connected magnetic scalar potential volume. In this case, either total or reduced potential will be multiply connected, since both have to represent the \mathbf{H}_e component of the field. Rather than producing a jump in the scalar potential as seen with multiply connected total potential regions enclosing non-zero currents, the scalar potential will force the net current in the **VECTOR** region to be zero to satisfy Ampere’s law.

To overcome this difficulty, the **VECTOR** region should be expanded to include some non-conducting material such that the net current is zero. In the case of the induction motor, this means that part of the laminated rotor between the bars should be specified as a **VECTOR** region. A simple example is shown in Figures 7.15 and 7.16, while in Figure 7.17 the **VECTOR** regions for a CARMEN model of an induction motor are shown. In the multiply connected model, the current vectors in the circuit show that both positive and negative currents are induced balancing to zero. The model has been corrected by making the region in the centre of the circuit **AIR VECTOR** and the current vectors now show unidirectional flow.

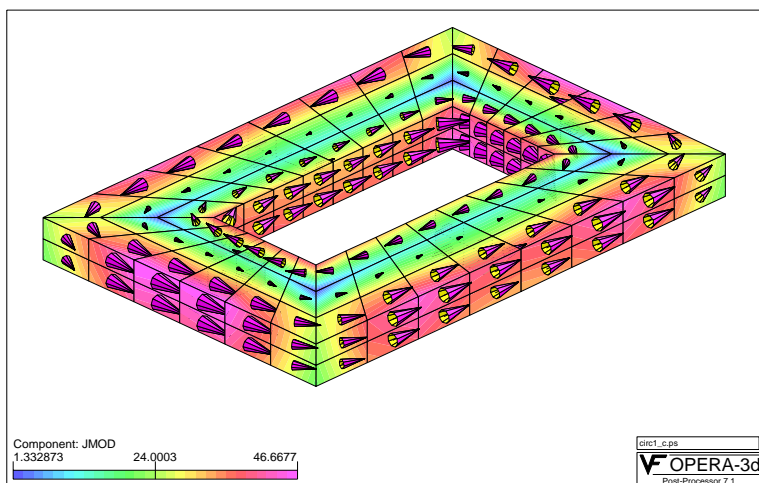


Figure 7.15 A multiply connected problem, showing wrong result if set up incorrectly

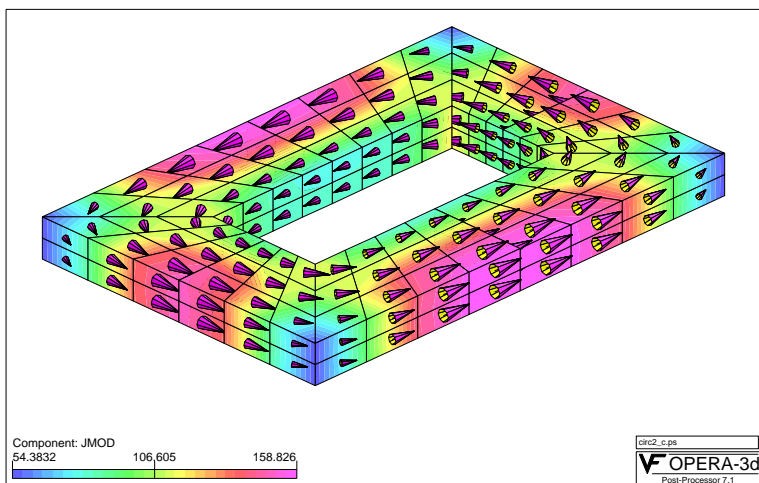


Figure 7.16 Corrected model with an **AIR VECTOR** region in the centre of the frame

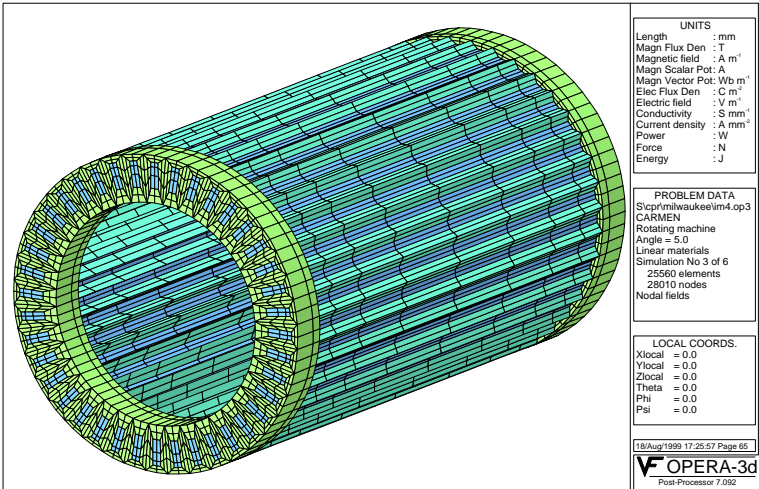


Figure 7.17 A multiply connected model in CARMEN showing the required **VECTOR** volumes

Power and Energy Calculation in AC Solutions

Power Calculations

If a current and voltage waveforms vary as:

$$\begin{aligned} i &= I \sin(\omega t) \\ v &= V \sin(\omega t) \end{aligned} \quad (7.17)$$

then the power in the system is given by:

$$\begin{aligned} P(\omega t) &= vi = VI \sin(\omega t) \sin(\omega t - \beta) \\ &= VI [\cos(\phi) - \cos(2\omega t - \phi)] \\ &= A - \frac{VI}{2} \cos(2\omega t - \phi) \\ &= A - \frac{VI}{2} \cos(\phi) \cos(2\omega t) - \frac{VI}{2} \sin(\phi) \sin(2\omega t) \\ &= A + B \cos(2\omega t) + C \sin(2\omega t) \end{aligned} \quad (7.18)$$

Hence when $\omega t=0$, then

$$\begin{aligned} P(0) &= A + B \\ P\left(\frac{\pi}{4}\right) &= A + C \\ P\left(\frac{\pi}{2}\right) &= A - B \end{aligned} \quad (7.19)$$

The time average power is given by:

$$P_{tav} = \frac{P(0) + P\left(\frac{\pi}{2}\right)}{2} = A = \frac{VI}{2} \cos(\phi) \quad (7.20)$$

Energy Calculations

Since non-linear materials are not modelled, in a magnetic circuit, **B** and **H** are sinusoidal. Replacing the *i* and *v* with **B** and **H**, the same arguments follow, such that the time average energy is given by:

$$\mathbf{E}_{tav} = \frac{\mathbf{E}(0) + \mathbf{E}\left(\frac{\pi}{2}\right)}{2} = A = \frac{\mathbf{B} \cdot \mathbf{H}}{2} \cos(\phi) \quad (7.21)$$

General

P is the system variable **POWER**, and E is the system variable **ENERGY**, set after the **ENERGY** command, and ωt is set by the **TIME** parameter. Further application notes are available under the relevant commands in the Reference Manual.

Inductance Calculations in OPERA-3d

The inductance of a coil is defined by either of the following equations:

$$E = \frac{1}{2}LI^2 \quad (7.22)$$

$$L = \frac{N\Phi}{I} \quad (7.23)$$

Where: E is the energy, I is the current applied at the terminals of the coil, Φ is the flux linking the coil, L is the inductance, and N is the number of turns in the coil.

Single Coil Problem

In problems where there is only one coil, no permanent magnets and no external fields, the inductance of the coil may be calculated in two ways.

No Magnetic Material Present

If no magnetic material is present (TOSCA and SCALA) or no magnetic or conducting material (ELEKTRA and CARMEN) is present, then the coil may be represented in the post processor with no mesh. The flux linking the coil may be calculated using a suitable patch and the expression for B_{normal} is integrated by the **MAP** command where

$$B_{normal} = B_x n_x + B_y n_y + B_z n_z \quad (7.24)$$

Then $\Phi = \iint B_{normal} ds$ and equation 7.23 may be used.

Magnetic Material Present

If a magnetic material is present (TOSCA and SCALA) or if a magnetic and/or conducting material is present (ELEKTRA and CARMEN), then a mesh must be prepared, and an analysis carried out. In the post processor, the total energy of the whole model may be calculated using the **ENERGY** command and the inductance is given by re-arranging equation 7.22.

Multiple Coil Problems – Linear Materials

In problems where linear magnetic or conductive material is present, and multiple coils are being modeled, the inductance can be determined in two ways.

Multiple Analyses And Superposition

The total inductance of the system is given by

$$L_{total} = L_{11} + M_{12} + M_{13} + \dots + M_{1n} + L_{22} + M_{21} + M_{23} + \dots + M_{2n} + L_{nn} + M_{1n} + M_{2n} + \dots + M_{nn} \quad (7.25)$$

$$L_{total} = L_{11} + L_{22} + \dots + 2M_{12} + 2M_{13} + \dots \quad (7.26)$$

To obtain L_{nn} an analysis is carried out with the n^{th} coil excited, and all other coils carrying no current. Each self-inductance is determined using a separate analysis and equation 7.22.

To obtain the mutual inductance, M_{nm} , an analysis is carried out with the m^{th} and n^{th} coils excited with all other coils carrying zero current. A second analysis with the n^{th} current reversed is then carried out. This gives two equations:-

$$E_1 = \frac{1}{2}L_{mm}I_m^2 + \frac{1}{2}L_{nn}I_n^2 + M_{nm}I_mI_n \quad (7.27)$$

$$E_2 = \frac{1}{2}L_{mm}I_m^2 + \frac{1}{2}L_{nn}I_n^2 - M_{nm}I_mI_n \quad (7.28)$$

Subtracting equations 7.27 and 7.28 gives

$$E_1 - E_2 = 2M_{nm}I_nI_m \quad (7.29)$$

and M_{nm} may be determined.

Flux Linkage

Using the flux, Φ , and the current in the coil, I , the self inductance can be calculated as:

$$L = \frac{N\Phi}{I} \quad (7.30)$$

A similar equation can be used for the mutual inductance, using the flux in one coil due to the flux generated by the current in another. This is calculated using:

$$M_{ij} = \frac{N_i\Phi_i}{I_j} \quad (7.31)$$

Using the solution to the model with one coil switched on will allow you to find the self inductance of the coil, and the mutual inductance of other coils with respect to that coil.

Multiple Coils – Non-linear Materials

If you wish to use non-linear properties it is necessary to have a solution with all coils at normal operating conditions. This gives the equation for the flux linking coil 1 as:

$$N_1 \Phi_1 = L_1 I_1 + M_{12} I_2 + M_{13} I_3 + \dots + M_{1n} I_n \quad (7.32)$$

for a model with n-coils. Similar equations can be generated for Φ_2 etc.

The model must then be altered so that the current in any one of the coils is changed by a small amount (ΔI), saving the model, and generating a new solution file (using restart run to speed matters up).

The small change should not greatly affect the field, but a small change of $\Delta \Phi_1$ will occur for each coil. For example, for a change of ΔI_1 :

$$\begin{aligned} N_1(\Phi_1 + \Delta \Phi_1) &= L_1(I_1 + \Delta I_1) + M_{12} I_2 + M_{13} I_3 + \dots + M_{1n} I_n \\ N_2(\Phi_2 + \Delta \Phi_2) &= L_2 I_2 + M_{21}(I_1 + \Delta I_1) + M_{23} I_3 + \dots + M_{2n} I_n \\ &\dots \end{aligned} \quad (7.33)$$

Hence taking the difference between equation 7.33a and equation 7.32, we can calculate L_1 :

$$L_1 = \frac{N_1 \Delta \Phi_1}{\Delta I_1} \quad (7.34)$$

Equation 7.33b and equation 7.32 will give M_{21} , and so forth. Modifying I_2 would allow similar calculations for L_2 etc. Therefore, finding the self and mutual inductance for all n coils requires n models to be solved.

This method can also be used if other energy sources are present, e.g. permanent magnets or external driving fields, as it works on the basis of a *change* in field.

A change in energy calculation is possible but would require more solutions to be calculated, as only one piece of information is found from each model.

Use of Command Scripts to calculate Fourier Series

It is possible to generate command scripts that will generate the different components of a Fourier series. These commands scripts make use of the \$ commands specified within the Reference manuals and are applicable to both 2D and 3D solutions.

The following is an example *.comi* file that calculates the Fourier components of **POTENTIAL** where the full period has been modelled. The commands are those for OPERA-2d but can be easily transferred to 3D by using the **LINE** command followed by a **PLOT** command within the main **DO** loop.

```
/ Set the number of Fourier components
$CONS #n 8
/ Set the line for the integral - for a straight line
$CONS #x1 0
$CONS #y1 0
$CONS #x2 10
$CONS #y2 0
/ Assume the line of integration covers
/ one whole period.
/ If the line only covers half a period -
/ set #symm to 2 and remove the odd or even
/ component from the fourier series (see later)
$CONS #symm 1
/ Get the length of the line and the fractional
/ distance along the line
/ If working on e.g. an arc section,
/ #T should be the total Angle subtended by the arc,
/ #X the fractional distance around this arc
$CONS #T SQRT((#x2-#x1)**2+(#y2-#y1)**2)
$PARA #X SQRT((x-#x1)**2+(y-#y1)**2)/(#T*#symm)
/ Set the function to be analysed
$PARA #Fx POT
/ Set the component that will be integrated for each
/ component
/ Temporarily set #i
$CONS #i 0
$PARA #Cosi Cos(2*PI*#I*#x)
$PARA #Sini Sin(2*PI*#I*#x)
/ Loop through each component
$DO #I 0 #n
/ Get the Cos Fourier components of the function #Fx
/ Not necessary if the function is Odd
INTL #x1 #y1 #x2 #y2 ERRO=128 COMP=(2/#T)*#Fx*#Cosi
$CONS #A%int(#I) INTEGRAL
/ Get the Sin Fourier components of the function #Fy
/ Not necessary if the function is Even
INTL #x1 #y1 #x2 #y2 ERRO=128 COMP=(2/#T)*#Fx*#Sini
$CONS #B%int(#I) INTEGRAL
$END DO
/ If #Fx is an Even function
$CONS #A0 #A0/2
```

The results of this analysis will be the components of the Fourier series described by

$$F(x) = a_0 + a_1 \cos(2\pi x t) + a_2 \cos(4\pi x t) + \dots \\ + b_1 \sin(2\pi x t) + b_2 \sin(4\pi x t) + \dots \quad (7.35)$$

or using the notation of the *.comi* file:

```
F(x) = #A0 + #A1*Cos(2*PI*#x/#T) + #A2*Cos(4*PI*#x/#T) + ...
... + #B1*Sin(2*PI*#x/#T) + #B2*Sin(4*PI*#x/#T) + ...
```

The *.comi* file above is a general purpose command file and must be modified for use in the correct context. If only part of the model has been created within the software, the commands above must be changed to reflect the partial nature of the results of the integral command, i.e.

F(x) is an Odd Function: $A_n = 0$

F(x) is an Even Function: $B_n = 0$

Only 1/4 period: Even harmonics e.g. A_2, A_4, B_2, B_4 are zero.

Complex Material Properties

Introduction

The introduction of complex material properties in a.c. solutions can be a powerful tool for modelling the physical behaviour of a material. The effect of a complex material property is to allow a phase difference between the two vector quantities in the constitutive relationships:

$$\mathbf{B} = \mu \mathbf{H}$$

$$\mathbf{D} = \epsilon \mathbf{E}$$

$$\mathbf{J} = \sigma \mathbf{E}$$

The most commonly used of these is a complex μ as a model for hysteresis. This has the effect of giving an elliptical relationship between magnetic flux density, \mathbf{B} , and magnetic field strength, \mathbf{H} . For example, a material specified with a relative permeability of 100 and a phase angle of 20 degrees ($\mu_r = 100 e^{-j(\pi/9)}$) gives a loop as shown. Complex permeability is available in ELEKTRA-SS and SOPRANO-SS.

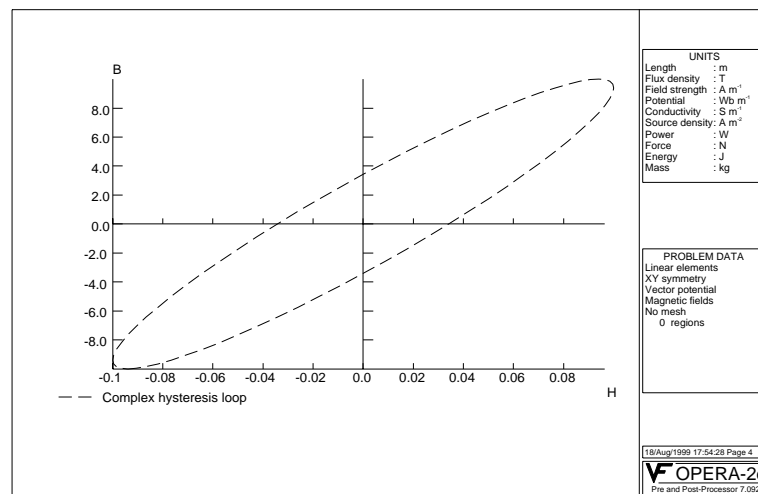


Figure 7.18 Typical hysteresis loop

A complex value of ϵ allows “lossy” dielectrics to be modelled, for example in the equation:

$$\nabla \cdot \epsilon^c \nabla V = 0$$

where $\epsilon^c = \epsilon' - j \epsilon''$, such that ϵ' is the permittivity of the material and $\sigma = \omega \epsilon''$. Complex permittivity is available in SOPRANO-SS and is useful when the lossy term is frequency dependent, rather than assigning independent values for ϵ and σ which will be used at all frequencies. It is also possible to use ELEKTRA-SS to model lossy dielectric materials. This is discussed later in this document.

Complex conductivity, σ , can be used in both ELEKTRA-SS and SOPRANO-SS. Its primary function is to represent the hysteretic effects associated with superconducting materials, although it can also be used to represent a lossy dielectric material (c.f. complex permittivity). In this case, the permittivity is frequency dependent.

Accessing Complex Materials in ELEKTRA-SS and SOPRANO-SS

Using a complex material property for 3D models is very simple. When the database (.op3 file) is being created, the **Materials** option on the **Analysis data** menu allows the user to specify the properties for each material name used in the model. For ELEKTRA-SS and SOPRANO-SS, phase angles (in degrees) may be specified. Note that if the **Multiple (anisotropic) material** options are chosen, the same phase angle is used in all three local co-ordinate directions.

Complex permeability in non-linear problems

When complex permeability is used in non-linear materials (ELEKTRA only), the choice of phase angle can become difficult. As shown in Figure 7.19, choosing a single phase angle that is applied over the whole **B** vs. **H** characteristic can give very different sized hysteresis loops. In fact, the hysteresis losses (and hence loop size) for a material operating at a peak flux density of 20 kGauss and at 25 kGauss will be nearly identical.

To obtain a more accurate solution, it is better to use several different phase angles for different operating regimes on the curve. This can be achieved as follows.

1. Run the model non-linearly without complex permeability
2. Determine the variation of peak flux density in the magnetic materials and discretise into a number of ranges. The shape of the **B** vs. **H** characteristic will dictate this discretisation but it is unlikely that all ranges will be equal. For example, the ranges (0 – 0.5), (0.5 – 1.0), (1.0 – 1.2), (1.2 – 1.3), (1.3 – 1.35) and (1.35 – 1.4) Tesla may be suitable for a material that is not very saturated.

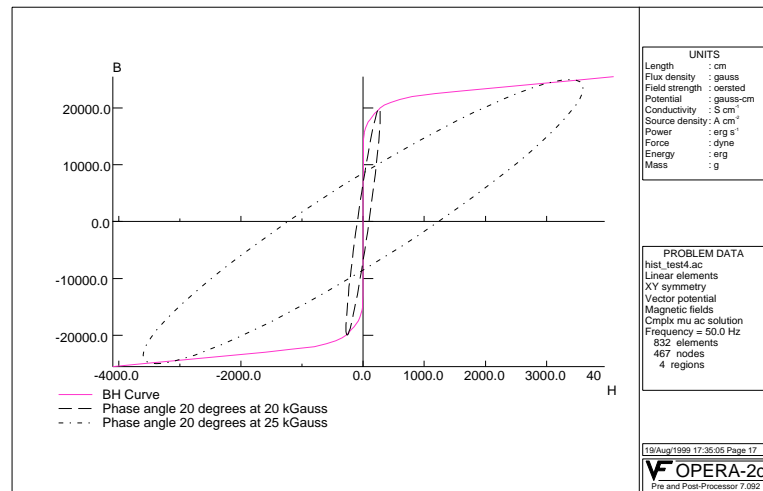


Figure 7.19 Using different phase angles for different parts of the BH curve

3. Re-assign the materials in the model so that a different material name is used for the parts of the model that fall into each range of flux density.
4. Run the new model non-linearly with complex permeability using the same **B** vs. **H** characteristic for all materials but assigning a different phase angle for each material name.

Using ELEKTRA-SS to model lossy dielectrics

When the complex permeability options in ELEKTRA are used in magnetic **TOTAL** scalar potential regions, the equation solved is:

$$\nabla \cdot \mu^c \nabla \psi = 0$$

which is identical in form to the equation for electric field problems given above. Consequently, it is possible to create a magnetic field problem that will behave numerically exactly as the electric field problem, and re-interpret the results displayed in the OPERA-3d post-processor. The value of the complex permittivity should be assigned to the permeability of the material and the conductivity assigned to zero. To assist in the re-interpretation of results, user defined parameters can be used.

```

/
/ Define parameters for electric scalar potential, #V,
/ electric field components, e.g. #EX,
/ electric flux density, e.g. #DX,
/ and current density, e.g. #JX
/
$PARAM #V POT

```

```

$PARAM #EX RHX*COST+IHX*SINT
$PARAM #EY RHY*COST+IHY*SINT
$PARAM #EZ RHZ*COST+IHZ*SINT
$PARAM #EMOD SQRT(#EX**2+#EY**2+#EZ**2)
$ASK #EPS 'Enter value of relative permittivity'
$ASK #SIG 'Enter value of conductivity (S/cm)'
$PARAM #DX #EPS*#EX
$PARAM #DY #EPS*#EY
$PARAM #DZ #EPS*#EZ
$PARAM #DMOD SQRT(#DX**2+#DY**2+#DZ**2)
$PARAM #JX #SIG*#EX
$PARAM #JY #SIG*#EY
$PARAM #JZ #SIG*#EZ
$PARAM #JMOD SQRT(#JX**2+#JY**2+#JZ**2)

```

It is recommended that CGS units be used throughout.

ELEKTRA-SS can, of course, also solve this problem using real **VOLTAGE** boundary conditions and a complex conductivity (as mentioned above) in **VECTOR** potential regions of the model. The disadvantage of this is that 3 components of vector potential, **A**, are also solved at every node.

OPERA-3D USER GUIDE TUTORIALS

Vector Fields Limited
24 Bankside
Kidlington
Oxford OX5 1JE
England

Copyright © 1999-2002 by Vector Fields Limited, England

This document was prepared using Adobe® FrameMaker®.

UNIX is a trademark of X/Open Company Ltd.

HPGL is a trademark of Hewlett-Packard Incorporated

PostScript is a trademark of Adobe Systems Incorporated

Windows 95/98/NT is a trademark of Microsoft Corporation

X Window System is a trademark of Massachusetts Institute of Technology

I-DEAS is a trademark of Structural Dynamics Research Corporation

OpenGL is a trademark of Silicon Graphics Corporation

All other brand or product names are trademarks or registered trademarks
of their respective companies or organisations.

Chapter 8

An Example Using the Pre Processor

Starting OPERA-3d Pre Processor

UNIX Operating Systems

In most cases it is advisable to run the programs from a suitable user directory rather than the directory containing the Vector Fields software. Similarly, it is also advisable to run the software as a “user” rather than as “system manager” status since this protects against accidental overwriting of files.

Hence, as a user from a suitable local directory, OPERA may be launched by entering:

opera

If both the OPERA-2d and OPERA-3d are installed, then the system prompts for a choice and the OPERA-3d should be selected:

```
2d or 3d processing or QUIT?  
3d
```

(If only OPERA-3d is been installed, then this choice is not given).

This is followed by a list of options relating to processing environments and analysis modules available and the pre processor should be selected:

```
Option:  
pre
```

If you have not set any environment variables in your system (see implementation notes) then you are requested to select a method of graphics display. In this case select the screen:

```
Graphics options: SCREEn, FILE, BOTH or NONE? >scre
```

A graphics window is then opened (in addition to the text window) and control moves to the menu system. Access to the menu system is from the main menu bar at the top of the graphics window.

Windows Systems

Launch the 3D pre processor from the OPERA Console or the Systray icon. The OPERA Console can be started with the supplied VF icon.

A simple model of an inductor

The first example using the pre processor is an enclosed inductor core with two air gaps. A cut away diagram is shown in Figure 8.1.

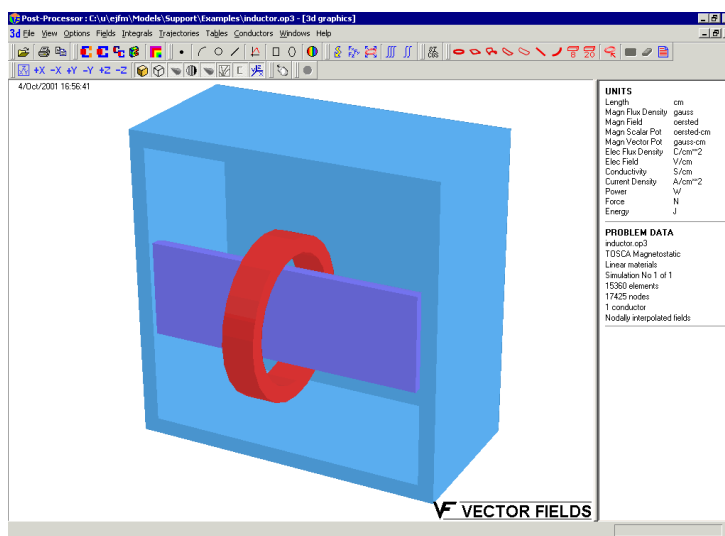


Figure 8.1 A section of the example inductor core

This example model will be created in the OPERA-3d pre processor and analysed using both TOSCA and ELEKTRA analysis modules. The results of this analysis will then be examined using the OPERA-3d post processor.

The model has symmetry, and only 1/8th of the model needs to be built. The complete model can be reconstructed later in the post processor.

The Baseplane of the Model

The model is built using a *Baseplane and Extrusion* method. This requires that a 2-dimensional section of the model is created. This is then extruded in the third dimension to build a 3-dimensional structure.

The method used to define the baseplane requires a series of steps:

1. Specify points in the baseplane that will be used to define the cross section geometry.
2. Use the points to create *facets*. These are surfaces which define the cross sectional geometry of the model.
3. Divide the facets to form a 2 dimensional finite element mesh.

Defining the baseplane points

Setting the coordinate system and axis size

Before starting to define the baseplane, switch on the 3d viewer. The 3d viewer is optional, and is not required for building a model.

```
DEFINE ↓
    Define new mesh → No 3d Viewer
                        (toggles to Show 3d view of the model)
```

By selecting the 3d viewer in this way, a separate window will pop up automatically after the first layer of elements has been created (see later).

To set the coordinate system of the baseplane, select

```
DEFINE ↓
    Define new mesh → Finite element mesh → XY plane,
                                                extrude in Z
```

This specifies that the baseplane will be in the XY plane and the extrusion direction is along the Z axis. To specify where on the Z axis the baseplane should be positioned, set

```
W coordinate of plane = 0
```

To enter the points to define the baseplane, an initial display size should be specified. To do this select

Minimum on horizontal axis	=	0
Maximum on horizontal axis	=	10
Minimum on vertical axis	=	0
Maximum on vertical axis	=	10
<input type="button" value="Accept"/>		<input type="button" value="Dismiss"/>

and **Accept** the settings. Then select **Point Input** from the **Define Baseplane** menu.

Entering construction lines

Construction lines may be used to define the geometry of the baseplane. These lines do not form part of the geometry but act as an aid to specifying coordinates. Specify the following construction lines to help with point definition.

... **Construction lines** → **Enter C_Lines** → **By parameters** → **Arc**

Arcs may be specified by centre point, start and finish points (in polar coordinates). To do this, specify the following:

Centre U	=	0
..... V	=	0
Start R	=	3
...Theta	=	0
End R	=	3
...Theta	=	90
<input type="button" value="Accept"/>		<input type="button" value="Dismiss"/>

and **Accept** the settings. Note that if the start and end radii are not the same, this will generate an archimedean spiral.

Add a second arc using:

Centre U	=	0
..... V	=	0
Start R	=	2
...Theta	=	0
End R	=	2
...Theta	=	90
<input type="button" value="Accept"/>		<input type="button" value="Dismiss"/>

Similarly, a number of straight construction lines are required and may be specified by start and finish points with an angle of rotation. Enter these using **Line**:

Start.	U	=	0
.....	V	=	0
Finish	U	=	10
.....	V	=	0
Rotation		=	0
<input type="button" value="Accept"/>		<input type="button" value="Dismiss"/>	

Start.	U	=	0
.....	V	=	10
Finish	U	=	10
.....	V	=	10
Rotation		=	0
<input type="button" value="Accept"/>		<input type="button" value="Dismiss"/>	

Start.	U	=	0
.....	V	=	9
Finish	U	=	10
.....	V	=	9
Rotation		=	0
<input type="button" value="Accept"/>		<input type="button" value="Dismiss"/>	

Start.	U	=	9
.....	V	=	0
Finish	U	=	9
.....	V	=	15
Rotation		=	0
<input type="button" value="Accept"/>		<input type="button" value="Dismiss"/>	

These lines are shown in Figure 8.2. Now **Return** three times to the **Point Definition** menu.

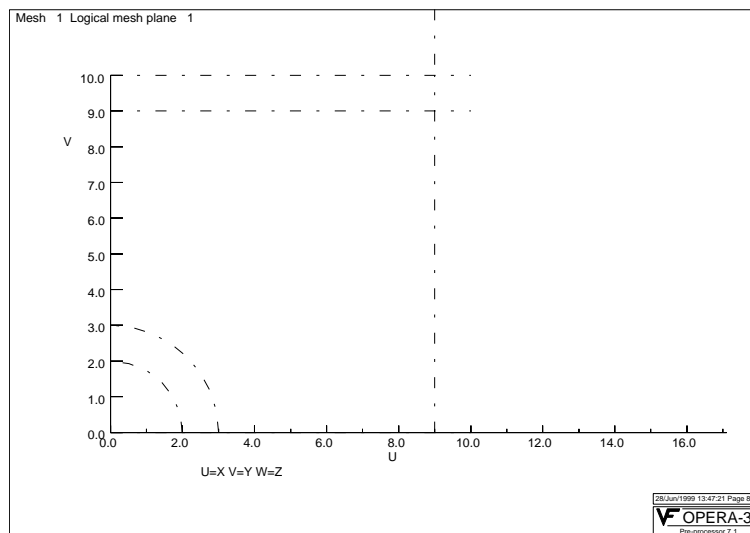


Figure 8.2 Display of construction lines

Entering baseplane points

The construction line intersections and line ends may be used to position the points required in the baseplane. Use **At C_line intersection** and follow this by using the cursor to *click on*¹ the ends and intersections of all the construction lines you have defined, i.e. at the following coordinates²:

```
0, 0
0, 2
0, 3
0, 9
0, 10
2, 0
3, 0
9, 0
9, 9
9, 10
10, 0
10, 9
10, 10
```

It is also possible to draw construction lines using existing points. To do this select

-
1. *Click on* means placing the cross cursor near the point and pressing the left mouse button. The cursor does not have to be placed precisely on the point
 2. The function key F1 or PF1 acts as a toggle switch to hide and reinstate the menus

```

... → Construction
      lines → Enter
           C_lines → By picking
                   points → Straight line

```

Select **Pick 2 points** and then *click on* the following points:

```

0,    0
10,   10

```

A construction line is drawn as shown in Figure 8.3.

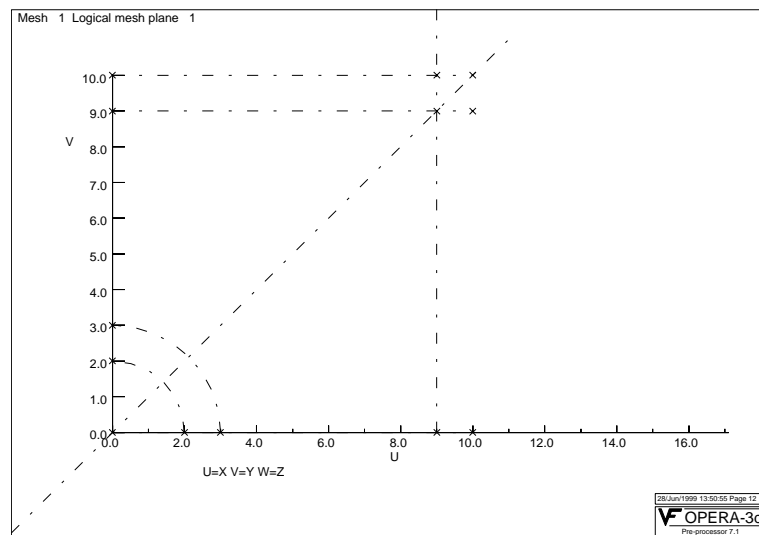


Figure 8.3 Display Showing Additional Points and Construction lines

Leave construction line input with **Return** three times. This will bring you back to the **Point Definition** menu.

Now we use this new construction line to place two points at the intersection by selecting

```

... At C_line intersection

```

and *click on* the points:

```

1.5,   1.5
2.4,   2.4

```

Points can also be defined explicitly in terms of their coordinate positions using cartesian or cylindrical polar coordinate systems. To enter points using cartesian coordinates select in the **Point Definition** menu

... Give U,V, W

and enter the following:

Cartesian Coordinate Input	
U Coordinate	<input type="text" value="1"/>
V Coordinate	<input type="text" value="1"/>
W Coordinate	<input type="text" value="0"/>
<input type="button" value="Accept"/>	<input type="button" value="Exit"/>

and **Accept**. Follow this with the next two points:

Cartesian Coordinate Input	
U Coordinate	<input type="text" value="1"/>
V Coordinate	<input type="text" value="0"/>
W Coordinate	<input type="text" value="0"/>
<input type="button" value="Accept"/>	<input type="button" value="Exit"/>

Cartesian Coordinate Input	
U Coordinate	<input type="text" value="0"/>
V Coordinate	<input type="text" value="1"/>
W Coordinate	<input type="text" value="0"/>
<input type="button" value="Accept"/>	<input type="button" value="Exit"/>

and **Exit** the cartesian coordinate input.

Points can also be defined explicitly in terms of their coordinate positions using cartesian or cylindrical polar coordinate systems. To enter points using cylindrical polar coordinates select in the **Point Definition** menu

... Give R, Theta, W

and enter

Polar Coordinate Input	
R Coordinate	<input type="text" value="2"/>
T Coordinate	<input type="text" value="22.5"/>
W Coordinate	<input type="text" value="0"/>
<input type="button" value="Accept"/>	<input type="button" value="Exit"/>

and **Accept**

Follow this with points at the following coordinates:

Polar Coordinate Input	
R Coordinate	<input type="text" value="3"/>
T Coordinate	<input type="text" value="22.5"/>
W Coordinate	<input type="text" value="0"/>
<input type="button" value="Accept"/>	<input type="button" value="Exit"/>

Polar Coordinate Input	
R Coordinate	<input type="text" value="3"/>
T Coordinate	<input type="text" value="22.5+45"/>
W Coordinate	<input type="text" value="0"/>
<input type="button" value="Accept"/>	<input type="button" value="Exit"/>

Polar Coordinate Input	
R Coordinate	<input type="text" value="2"/>
T Coordinate	<input type="text" value="67.5"/>
W Coordinate	<input type="text" value="0"/>
<input type="button" value="Accept"/>	<input type="button" value="Exit"/>

This gives the display shown in Figure 8.4.

Select **Exit** to leave the polar coordinate input. Press **Return** to close the **Point Definition** menu.

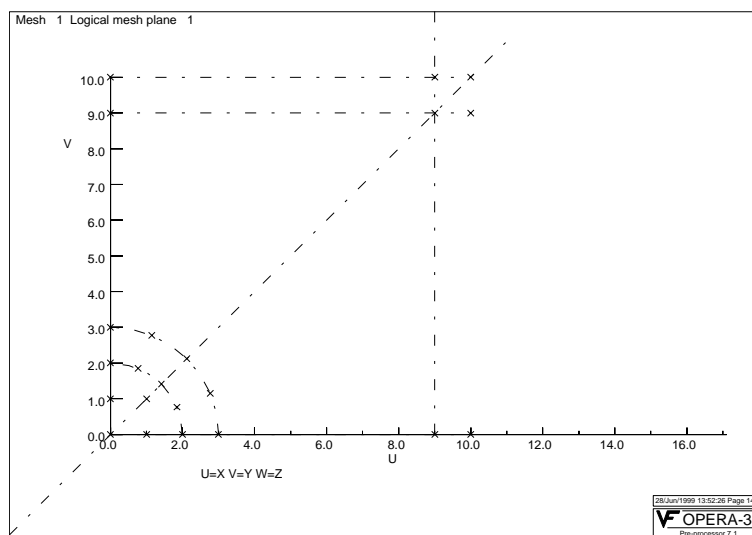


Figure 8.4 Display Showing the Polar Coordinate Points

Defining the baseplane facets

The baseplane points have now been positioned and these are used to form the baseplane facets. To do this select **Facet Input** from the **Define Baseplane** menu.

The **Facet Definition** sub-menu is displayed. The simplest facet to define consists of four corner points. Select **auto-close after 4** and *click on* the following points:

0, 10
 9, 10 (this creates Facet 1)
 9, 9
 0, 9

to complete the first facet. This is shown in Figure 8.5¹

Continue to create facets by selecting **auto-close after 4** and *click on* the following points:

9, 10
 10, 10 (Facet 2)
 10, 9
 9, 9

1. N.B. If you create an incorrect facet, you can delete it using **Delete facet**

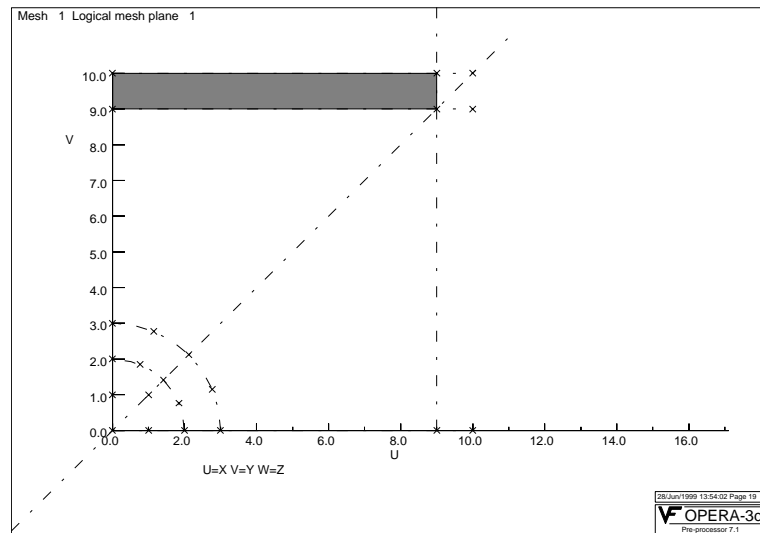


Figure 8.5 Display of the first facet.

9, 9
 10, 9 (Facet 3)
 10, 0
 9, 0

0, 1
 1, 1 (Facet 4)
 1, 0
 0, 0

To create facets with curved edges, points should be used as mid-side nodes. To do this select **no auto-close** and *click on* the following points:

9, 0
 9, 9 Three corners of Facet 5
 2.4, 2.4

Select **Mid-side** and *click on*

3, 1.2 Mid-side node creates curved edge of Facet 5

Select **no auto-close** and *click on*

3, 0 Final corner of Facet 5

and close the facet with **Close**.

Continue by using the following menu selections and points:

no auto-close	2.4, 2.4	
	9, 9	
	0, 9	Facet 6
	0, 3	
Mid-side	1.2, 2.7	

Note how the facet is closed automatically when a mid-side point is selected for the fourth side.

no auto-close	0, 3	
	0, 2	
Mid-side	0.7, 1.9	
no auto-close	1.5, 1.5	Facet 7
	2.4, 2.4	
Mid-side	1.2, 2.7	

no auto-close	2.4, 2.4	
	1.5, 1.5	
Mid-side	1.9, 0.8	
no auto-close	2, 0	Facet 8
	3, 0	
Mid-side	2.8, 1.2	

no auto-close	2, 0	
	1, 0	
	1, 1	Facet 9
	1.5, 1.5	
Mid-side	1.9, 0.8	

no auto-close	0, 2	
	0, 1	
	1, 1	Facet 10
	1.5, 1.5	
Mid-side	0.8, 1.9	

This completes the facet definition. The display of all the baseplane facets is shown in Figure 8.6. Press **Return** to close the **Facet Definition** menu.

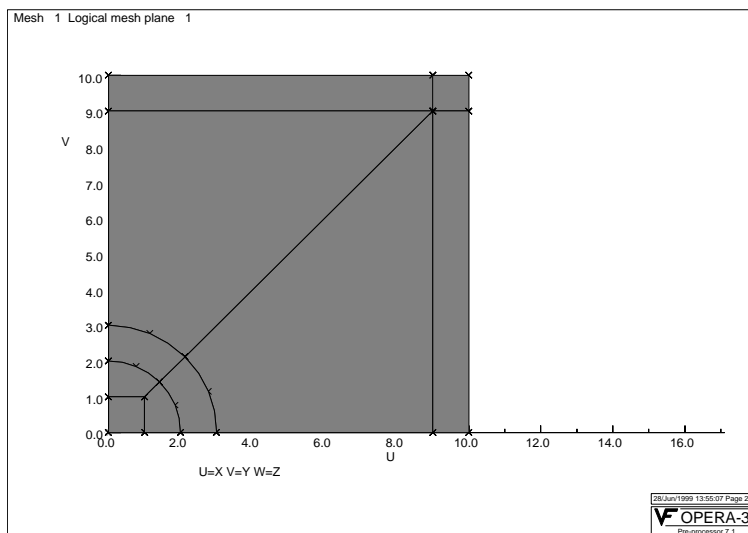


Figure 8.6 Display of all the facets

Defining the baseplane subdivisions

Move to the subdivision definition menu by selecting **Subdivision** from the **Define Baseplane** menu.

The baseplane subdivisions define the distribution of finite elements in the baseplane. These may be defined globally (the same value over the whole baseplane) or on individual lines (facet edges) or both.

Set the number of subdivisions to 8 by selecting **Set subdivision**.

Then *click on* button 8 and **Accept**.

Subdivision		
<input type="checkbox"/> 1	<input type="checkbox"/> 2	<input type="checkbox"/> 3
<input type="checkbox"/> 4	<input type="checkbox"/> 5	<input type="checkbox"/> 6
<input type="checkbox"/> 7	<input checked="" type="checkbox"/> 8	<input type="checkbox"/> 9
Other	<input type="text"/>	
<input type="button" value="Accept"/>		

Now select **Apply globally** to give all facets edges 8 divisions¹. The display indicates the distribution of subdivisions. Clear the message box displayed and select **Return** twice to return to the **Define Baseplane** menu.

Extending to the Third Dimension

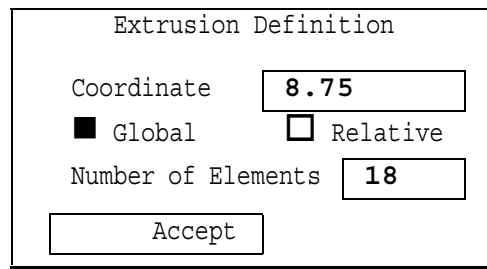
Defining extrusions in the third dimension

Once the baseplane has been completed, the third dimension is defined by sweeping the baseplane through space in the third dimension. To move to the extrusion definition, select

Extrude

and select **Linear extrusion**. Note that once started extruding, it is not possible to go back to edit the baseplane.

Complete the subsequent dialog box as shown below. This sets the first extrusion to a coordinate of 8.75 with 18 subdivisions in the extrusion direction.



The dialog box titled "Extrusion Definition" contains the following fields and options:

- Coordinate:** A text box containing the value "8.75".
- Global:** A radio button that is selected (indicated by a filled square).
- Relative:** An unselected radio button (indicated by an empty square).
- Number of Elements:** A text box containing the value "18".
- Accept:** A button at the bottom of the dialog.

Accept this dialog box and clear the message box displayed. Select **Finish Editing** followed by **Finish** four times to close the submenus (materials and boundary conditions will be defined later). The first extrusion is now completed.

Note that the since the 3d-Viewer was previously activated, a window now pops up after having created the first layer of elements.

Return once to the top-level menu.

Extend the extrusion further by selecting

DEFINE ↓

Extend existing mesh → **Extend without editing**

and **Accept** Mesh number = 1.

1. Note that this method ensures all facet edges have subdivision defined and that the mesh is continuous

Select **XY plane**, **extrude in Z** from the coordinate system submenu to continue extruding in the same direction. To create a second linear extrusion complete the subsequent dialog box as follows:

Extrusion Definition	
Coordinate	<input type="text" value="9"/>
<input checked="" type="checkbox"/> Global	<input type="checkbox"/> Relative
<input checked="" type="checkbox"/> Linear	<input type="checkbox"/> Quadratic
Number of Elements	<input type="text" value="2"/>
Accept - and extend again	
<input type="text" value="Accept - this is the last"/>	

accepting the extrusion with **Accept – and extend again**. Repeat the procedure to extend the model to coordinate 10 with 4 subdivisions but this time use **Accept – this is the last**.

Extrusion Definition	
Coordinate	<input type="text" value="10"/>
<input checked="" type="checkbox"/> Global	<input type="checkbox"/> Relative
<input checked="" type="checkbox"/> Linear	<input type="checkbox"/> Quadratic
Number of Elements	<input type="text" value="4"/>
<input type="text" value="Accept - and extend again"/>	
Accept - this is the last	

Return twice to the top-level menu.

The 3d-Viewer contains a wire-frame view of the model. The options of the 3d-Viewer will be discussed later in this chapter after having defined materials, boundaries and the conductor.

Objects in the 3D Model

Defining materials and potentials

The model is built up of different materials. These are used to describe the objects or parts of the model. So far, the whole model is made up of the default material – **AIR**.

It is also necessary to define the type of magnetic potential to be used for each material. For a detailed explanation of the choice of potentials please refer to the OPERA-3d Reference Manual. In the present model, although **TOTAL** potential is appropriate, an ELEKTRA solution will later be obtained using the same model definition. It is necessary therefore to label some materials as **VECTOR** potential, ready for ELEKTRA. In TOSCA, these will be interpreted as **TOTAL** scalar potentials.

To change the material definition from the default setting i.e. **AIR** and **TOTAL** potential, select

MODIFY ↓

Material properties

and **Accept** Layer number = 1.

On the display, the facets represent the volumes in the first layer of the model.

From the **Materials** submenu select **Select and define** and *click on* any volume. A material definition dialog box is displayed, which should be completed as follows:

- Material name: **AIR**
- Potential type: **REDUCED SCALAR**
- Element type: **LINEAR**
- Other layers¹: **FROM 1 TO ***
- **ALL VOLUMES**

1. * implies the highest value available i.e. maximum value or all layers.

The complete dialog box should look like this:

Material Definition			
Material Name	<input type="text" value="air"/>		
Potential Type:	Element Type:		
<input type="checkbox"/> Total Scalar	<input checked="" type="checkbox"/> Linear		
<input checked="" type="checkbox"/> Reduced Scalar	<input type="checkbox"/> Quadratic		
<input type="checkbox"/> Vector			
Options:			
Jx, Jy, Jz	<input type="text"/>		
Vx, Vy, Vz	<input type="text"/>		
Scalar: Charge Density or Rotational Velocity			
Scalar	<input type="text"/>		
Packing factor	<input type="text"/>		
Material orientation			
<input type="checkbox"/> Local XYZ=XYZ	<input type="checkbox"/> Local XYZ=YZX	<input type="checkbox"/> Local XYZ=ZXY	
Other vector	<input type="text"/>		
Other volumes and layers:			
From	<input type="text" value="1"/>	To	<input type="text" value="*"/> <input checked="" type="checkbox"/> All volumes
<input type="button" value="Accept"/>	<input type="button" value="Keep"/>	<input type="button" value="Help"/>	<input type="button" value="Quit"/>

Accept the settings. This redefines all regions to be AIR and REDUCED scalar potential. This is the correct setting for regions containing source conductors.

Objects in the model are defined by their material names. To change from the **AIR** definition (from the **Materials** submenu) select

Select/de-select volume and select volumes by using the cursor to *click on* all but one of the volumes which form the central cylindrical part of the model. Use the following coordinates ¹

0.5, 0.5
 0.5, 1.5
 1.0, 2.5
 1.5, 0.5

and then select **Select and define** and *click on* the last volume in the central part at coordinate

1. If an incorrect volume is selected, *click on* the volume again to deselect it.

2.5, 1.0

Again a material definition dialog box is displayed. Complete this material name, potential and elements types as follows

Material Definition

Material Name

Potential Type: ☐ Total Scalar ☒ Reduced Scalar ☒ Vector

Element Type: ☒ Linear ☐ Quadratic

Options:

Jx, Jy, Jz

Vx, Vy, Vz

Scalar: Charge Density or Rotational Velocity

Scalar

Packing factor

Material orientation

☐ Local XYZ=XYZ ☐ Local XYZ=YZX ☐ Local XYZ=ZXY

Other vector

Other volumes and layers:

From To ☐ All volumes

and **Accept** the settings. This defines the central core of the device which should now be displayed in blue. Although **VECTOR** potential is selected, in TOSCA this will be interpreted as **TOTAL** scalar potential. Repeat the procedure for the facets which form the box using the following coordinate with **Select/de-select volume**

5.0, 9.5
9.5, 9.5

and **Select and define**

9.5, 5.0

Complete the subsequent dialog box with material name, potential and element types and layer numbers as shown.

Material Definition			
Material Name	<input type="text" value="box"/>		
Potential Type:	Element Type:		
<input type="checkbox"/> Total Scalar	<input checked="" type="checkbox"/> Linear		
<input type="checkbox"/> Reduced Scalar	<input type="checkbox"/> Quadratic		
<input checked="" type="checkbox"/> Vector			
Options:			
Jx, Jy, Jz	<input type="text"/>		
Vx, Vy, Vz	<input type="text"/>		
Scalar: Charge Density or Rotational Velocity			
Scalar	<input type="text"/>		
Packing factor	<input type="text"/>		
Material orientation			
<input type="checkbox"/> Local XYZ=XYZ	<input type="checkbox"/> Local XYZ=YZX	<input type="checkbox"/> Local XYZ=ZXY	
Other vector	<input type="text"/>		
Other volumes and layers:			
From	<input type="text" value="1"/>	To	<input type="text" value="2"/> <input type="checkbox"/> All volumes
<input type="button" value="Accept"/>	<input type="button" value="Keep"/>	<input type="button" value="Help"/>	<input type="button" value="Quit"/>

This defines the sides of the outer case which should be displayed in a lighter shade of blue. To further see the effect of this (from the **Materials** submenu) select **Show properties**.

The material properties are displayed as shown in Figure 8.7. Select **Finish** to close the submenu.

To complete the material definition, select

MODIFY ↓

Material properties

and

Accept Layer number = 3.

From the **Materials** submenu select **Select and define** and *click on* any volume. Complete the dialog box as follows to change **ALL** the volumes in this layer:

Material Definition			
Material Name	<input type="text" value="box"/>		
Potential Type:	Element Type:		
<input type="checkbox"/> Total Scalar	<input checked="" type="checkbox"/> Linear		
<input type="checkbox"/> Reduced Scalar	<input type="checkbox"/> Quadratic		
<input checked="" type="checkbox"/> Vector			
Options:			
Jx, Jy, Jz	<input type="text"/>		
Vx, Vy, Vz	<input type="text"/>		
Scalar: Charge Density or Rotational Velocity			
Scalar	<input type="text"/>		
Packing factor	<input type="text"/>		
Material orientation			
<input type="checkbox"/> Local XYZ=XYZ	<input type="checkbox"/> Local XYZ=YZX	<input type="checkbox"/> Local XYZ=ZXY	
Other vector	<input type="text"/>		
Other volumes and layers:			
From	<input type="text"/>	To	<input type="text"/> <input checked="" type="checkbox"/> All volumes
<input type="button" value="Accept"/>	<input type="button" value="Keep"/>	<input type="button" value="Help"/>	<input type="button" value="Quit"/>

and **Accept**. This defines the top of the outer case. To see the effect of this (from the **Materials** submenu) select **Show properties**.

The material properties are displayed as shown in Figure 8.8.

Select **Finish** to close the submenu.

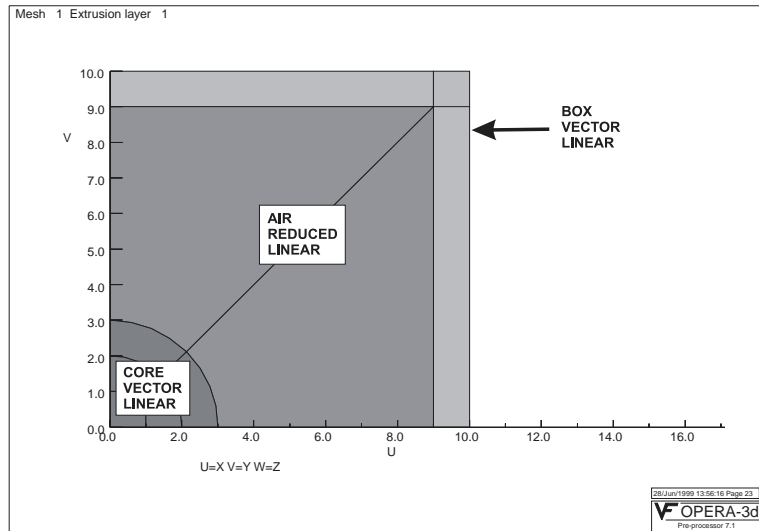


Figure 8.7 Display of material properties in Layer 1

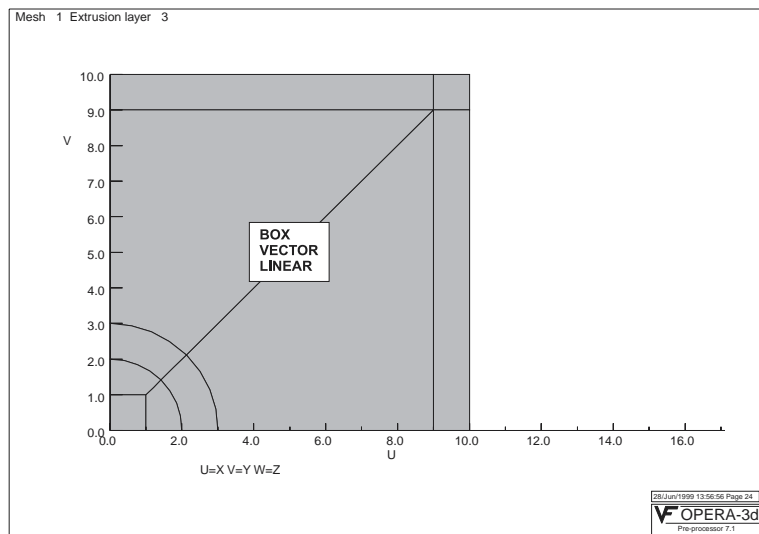


Figure 8.8 Display of material properties in Layer 3

Magnetic Boundary Conditions

Defining boundary conditions

The magnetic field conditions on the model boundaries should now be specified.
To do this select

MODIFY ↓

Boundary conditions → Base plane → Select and define

and *click on* any facet. Complete the subsequent dialog box as follows to apply the **NORMAL MAGNETIC** condition to **ALL** facets:

Boundary Conditions			
Condition name:			
<input type="checkbox"/> Magnetic Scalar	<input checked="" type="checkbox"/> Normal Magnetic	<input type="checkbox"/> Tangential Magnetic	
<input type="checkbox"/> Voltage	<input type="checkbox"/> Normal Electric	<input type="checkbox"/> Tangential Electric	
<input type="checkbox"/> Total Ax	<input type="checkbox"/> Total Ay	<input type="checkbox"/> Total Az	
<input type="checkbox"/> Incident Ax	<input type="checkbox"/> Incident Ay	<input type="checkbox"/> Incident Az	
<input type="checkbox"/> Incident Voltage	<input type="checkbox"/> Perfect Conductor	<input type="checkbox"/> Radiation	
<input type="checkbox"/> Normal Derivative	<input type="checkbox"/> Mixed Derivative		
<input type="checkbox"/> Symmetry	<input type="checkbox"/> Slip Surface	<input type="checkbox"/> Clear	
Value	<input type="text"/>	Label/2nd value	<input type="text"/>
Other volumes and layers:			
From	<input type="text"/>	To	<input type="text"/> <input checked="" type="checkbox"/> All facets
<input type="button" value="Accept"/>		<input type="button" value="Keep"/>	
<input type="button" value="Help"/>		<input type="button" value="Quit"/>	

and **Accept**.

This sets the magnetic field to be normal to the whole of the base plane and is required to give the correct reflection symmetry to the model. Select **Finish** to close the submenu.

All the other outer surfaces of the model need the **TANGENTIAL MAGNETIC** condition. To define the boundary conditions on the outer top surface of the box select

...Top plane → Select and define

and *click on* any facet. Complete the subsequent dialog box as follows

Boundary Conditions		
Condition name:		
<input type="checkbox"/> Magnetic Scalar	<input type="checkbox"/> Normal Magnetic	<input checked="" type="checkbox"/> Tangential Magnetic
<input type="checkbox"/> Voltage	<input type="checkbox"/> Normal Electric	<input type="checkbox"/> Tangential Electric
<input type="checkbox"/> Total Ax	<input type="checkbox"/> Total Ay	<input type="checkbox"/> Total Az
<input type="checkbox"/> Incident Ax	<input type="checkbox"/> Incident Ay	<input type="checkbox"/> Incident Az
<input type="checkbox"/> Incident Voltage	<input type="checkbox"/> Perfect Conductor	<input type="checkbox"/> Radiation
<input type="checkbox"/> Normal Derivative	<input type="checkbox"/> Mixed Derivative	
<input type="checkbox"/> Symmetry	<input type="checkbox"/> Slip Surface	<input type="checkbox"/> Clear
Value	<input type="text"/>	Label/2nd value <input type="text"/>
Other volumes and layers:		
From	<input type="text"/>	To <input type="text"/> <input checked="" type="checkbox"/> All facets
<input type="button" value="Accept"/>	<input type="button" value="Keep"/>	<input type="button" value="Help"/> <input type="button" value="Quit"/>

and **Accept**.

This sets the magnetic field to be tangential to the whole of this plane and implies that there is no flux leakage from the box. Select **Finish** to close the submenu.

To define the boundary conditions on the extruded facets select **Extrusion facets**.

Then set Layer number = 1 and **Accept**.

Select **Select/de-select facet**. In the picture the *lines* represent *facets* orthogonal to the plane of the screen. Select each of the facets around the outside of the model by *clicking on* the following coordinates:

0.1, 0.5
 0.1, 1.5
 0.1, 2.5
 0.1, 6.0
 0.1, 9.5
 5.0, 9.9
 9.5, 9.9
 9.9, 9.5
 9.9, 5.0
 0.5, 0.1
 1.5, 0.1

2.5, 0.1
9.5, 0.1

Then select **Select and define** and *click on* coordinate:

5.0, 0.1

Complete the subsequent dialog box to apply the boundary condition to all layers as follows:

Boundary Conditions		
Condition name:		
<input type="checkbox"/> Magnetic Scalar	<input type="checkbox"/> Normal Magnetic	<input checked="" type="checkbox"/> Tangential Magnetic
<input type="checkbox"/> Voltage	<input type="checkbox"/> Normal Electric	<input type="checkbox"/> Tangential Electric
<input type="checkbox"/> Total Ax	<input type="checkbox"/> Total Ay	<input type="checkbox"/> Total Az
<input type="checkbox"/> Incident Ax	<input type="checkbox"/> Incident Ay	<input type="checkbox"/> Incident Az
<input type="checkbox"/> Incident Voltage	<input type="checkbox"/> Perfect Conductor	<input type="checkbox"/> Radiation
<input type="checkbox"/> Normal Derivative	<input type="checkbox"/> Mixed Derivative	
<input type="checkbox"/> Symmetry	<input type="checkbox"/> Slip Surface	<input type="checkbox"/> Clear
Value	<input type="text"/>	Label/2nd value <input type="text"/>
Other volumes and layers:		
From	<input type="text" value="1"/>	To <input type="text" value="*"/> <input type="checkbox"/> All facets
<input type="button" value="Accept"/>	<input type="button" value="Keep"/>	<input type="button" value="Help"/> <input type="button" value="Quit"/>

Accept and this sets the magnetic field to be tangential on the outer surfaces of the box and implies that there is no flux leakage from the box. Select **Finish** to close the submenu followed by **Return** twice. This completes the boundary condition definition of the model.

Displaying the 3D Model

To examine the 3D model select

DISPLAY ↓

Display Command... ...view

and complete the parameter box:

Display View			
Size	<input type="text" value="10"/>		
Eye position:			
X	<input type="text" value="1"/>	Y	<input type="text" value="1"/>
Z	<input type="text" value="1"/>		
Centre of picture:			
X	<input type="text" value="0"/>	Y	<input type="text" value="0"/>
Z	<input type="text" value="0"/>		
Rotate picture	<input type="text" value="0"/>		
<input checked="" type="checkbox"/> New picture	<input type="checkbox"/> Add to picture		
<input checked="" type="checkbox"/> Parallel view	<input type="checkbox"/> Perspective view		
<input checked="" type="checkbox"/> Show Axes	<input type="checkbox"/> No axes		
<input type="button" value="Refresh Display"/>	<input type="button" value="Accept"/>	<input type="button" value="Quit"/>	

and **Refresh display**.

A wire frame display of the model is produced. Change the parameter box settings to give the opposite viewing x-direction as follows

Display View			
Size	<input type="text" value="10"/>		
Eye position:			
X	<input type="text" value="-1"/>	Y	<input type="text" value="1"/>
Z	<input type="text" value="1"/>		
Centre of picture:			
X	<input type="text" value="0"/>	Y	<input type="text" value="0"/>
Z	<input type="text" value="0"/>		
Rotate picture	<input type="text" value="0"/>		
<input checked="" type="checkbox"/> New picture	<input type="checkbox"/> Add to picture		
<input checked="" type="checkbox"/> Parallel view	<input type="checkbox"/> Perspective view		
<input checked="" type="checkbox"/> Show Axes	<input type="checkbox"/> No axes		
<input type="button" value="Refresh Display"/>	<input type="button" value="Accept"/>	<input type="button" value="Quit"/>	

and **Refresh display** again. Select **Return** to close the submenu.

To view the model with hidden surfaces removed, it is first necessary to construct the surface mesh. This is done using:

MESH ↓
 ...quadrilaterals

Clear the message box, and select **Return**.

To now display the model, select

DISPLAY ↓
 Display Command... ..style

and select the following settings of the subsequent dialog box:

Display Style	
Line view	<input type="checkbox"/>
Surface view	<input checked="" type="checkbox"/>
Full surface algorithm	<input type="checkbox"/>
<hr/>	
No Elements	<input type="checkbox"/>
Surface Elements	<input checked="" type="checkbox"/>
Volume Elements	<input type="checkbox"/>
<hr/>	
Vectors...	<input type="checkbox"/>
... no vectors	<input checked="" type="checkbox"/>
... in conductors only	<input type="checkbox"/>
... material orientation	<input type="checkbox"/>
... current density	<input type="checkbox"/>
... velocity	<input type="checkbox"/>
<hr/>	
Refresh display	*
Return	←

Finish by selecting **Refresh display**.

Return twice to close the submenus. The model with the discretisation of the mesh is shown in Figure 8.9.

The final stage is to create the volume mesh. This is completed using:

MESH ↓
 Mesh

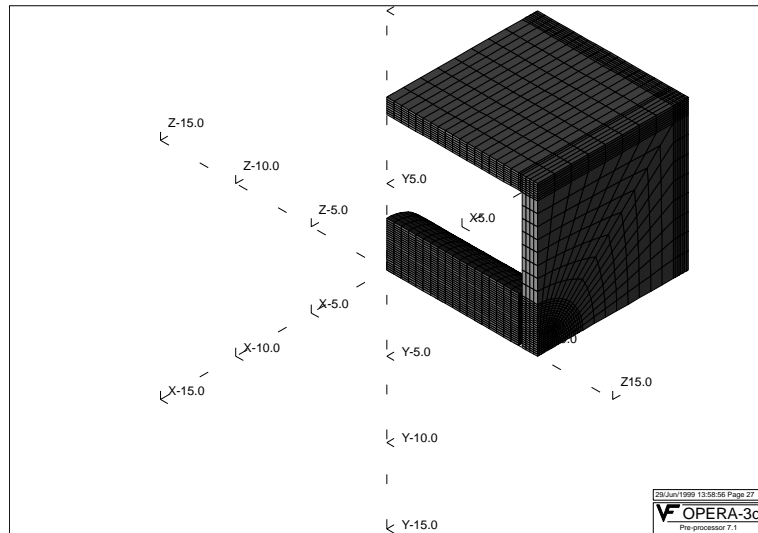


Figure 8.9 Display of 3D finite element mesh

and **Accept** to accept the default parameters. On successful completion of the volume mesh, clear the message box.

If desired, the volume mesh can also be displayed, but for large models it is warned that this can be a slow procedure.

Defining the Conductor

The conductors in OPERA-3d do not form part of the finite element mesh but are assumed to be source conductors. They are defined separately from the mesh. To do this select

DEFINE ↓

Conductors → Define a
conductor → Generally
orientated set → Solenoid

A sequence of parameter boxes and menus is now displayed. Complete them and **Accept** as follows:

Local coord 1: X - origin	=	0
Local coord 1: Y - origin	=	0
Local coord 1: Z - origin	=	0
Accept		

Local Coordinate system 1	
XYZ local = XYZ global	<input checked="" type="checkbox"/>
XYZ local = YZX global	<input type="checkbox"/>
XYZ local = ZXY global	<input type="checkbox"/>
Other system	<input type="checkbox"/>
Return	←

Local coord 2: X - origin	=	0
Local coord 2: Y - origin	=	0
Local coord 2: Z - origin	=	0
Accept		

Local Coordinate system 2	
XYZ local = XYZ global	<input type="checkbox"/>
XYZ local = YZX global	<input checked="" type="checkbox"/>
XYZ local = ZXY global	<input type="checkbox"/>
Other system	<input type="checkbox"/>
Return	←

Cross-section: X1	=	4
Cross-section: Y1	=	-1
Cross-section: X2	=	5
Cross-section: Y2	=	-1
<input type="button" value="Accept"/>		

Cross-section: X3	=	5
Cross-section: Y3	=	1
Cross-section: X4	=	4
Cross-section: Y4	=	1
<input type="button" value="Accept"/>		

Curvature CU1	=	0
Curvature CU2	=	0
Curvature CU3	=	0
Curvature CU4	=	0
<input type="button" value="Accept"/>		

Current Density	=	1000
Symmetry code	=	1
Drive label	=	ONE
<input type="button" value="Accept"/>		

Conductor reflections	
No reflection in XY(1)	<input checked="" type="checkbox"/>
+ reflection in XY(1)	<input type="checkbox"/>
- reflection in XY(1)	<input type="checkbox"/>
No reflection in YZ(1)	<input checked="" type="checkbox"/>
+ reflection in YZ(1)	<input type="checkbox"/>
- reflection in YZ(1)	<input type="checkbox"/>
No reflection in ZX(1)	<input checked="" type="checkbox"/>
+ reflection in ZX(1)	<input type="checkbox"/>
- reflection in ZX(1)	<input type="checkbox"/>
Return	*

Tolerance on flux density	=	0.0001
<input type="button" value="Accept"/>		

Select **Return** four times to close all the menus and select

DISPLAY ↓

Display command... ..refresh display

to see the super-position of the solenoid on the finite element mesh. This is shown in Figure 8.10.

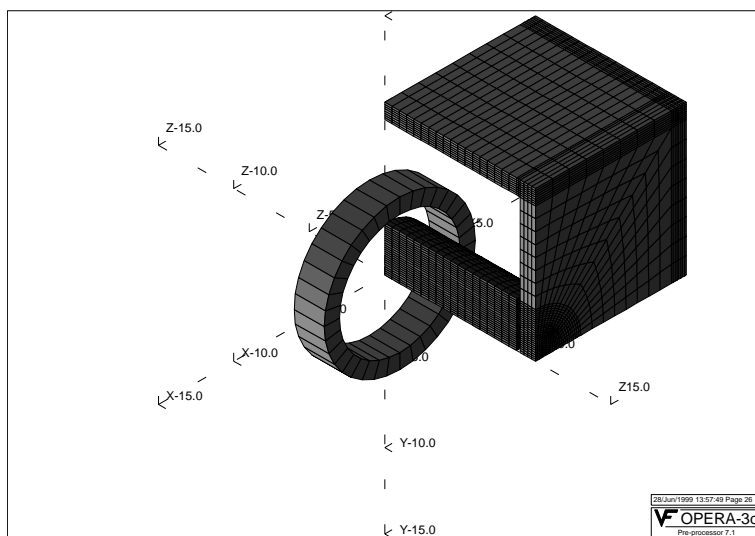


Figure 8.10 Display of mesh with solenoid

Examining the model with the 3d viewer

On a machine with OpenGL support the model can be examined with the “3d Viewer”. In this worked example the 3d Viewer is being used to examine the position of the conductor.

```

DISPLAY ↓
    3d Viewer...
        ...style
        Surface elements
DISPLAY ↓
    3d Viewer...
        ...refresh display
  
```

This will bring up an additional window in the pre processor window. Depending on your installation this new 3d Viewer window may overlap your existing Graphics Window in the pre processor window. Resize and move the new 3d Viewer window to any suitable location. Figure 8.11 shows a screenshot of a Windows installation, where the 3d Viewer is overlapping a part of the Graphics Window.

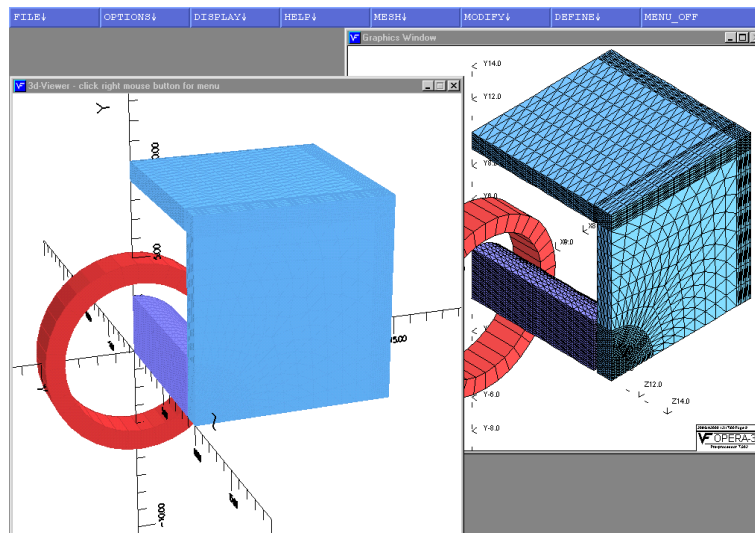


Figure 8.11 3d Viewer after initialisation

The view within the 3d Viewer can be manipulated using the mouse. The main control is the left mouse button. Moving the mouse with the left button pressed changes the view of the model. The way in which the view is changed and other options can be selected from a menu which pops up when the right mouse button is clicked.

After initialisation, the **Rotate** menu option is selected by pressing the right mouse, and selecting **Rotate** from the menu.

Rotate the model to obtain a view similar to the screenshot in Figure 8.12 by holding the left mouse button, and dragging the mouse.

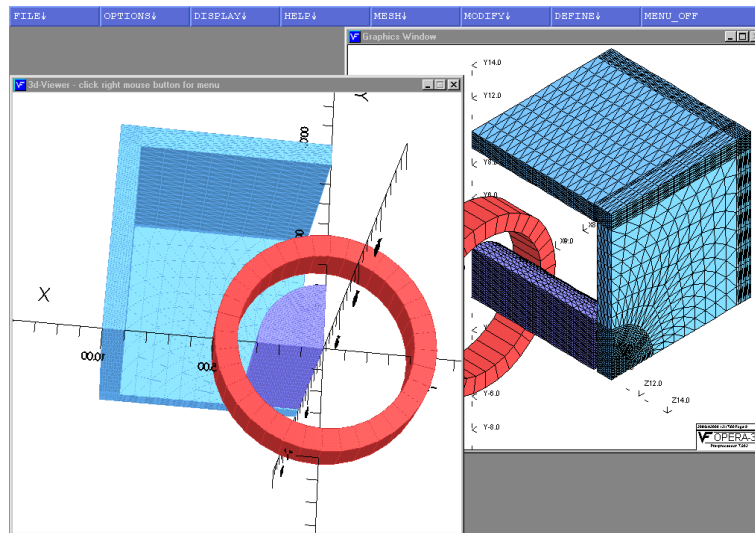


Figure 8.12 3d Viewer after rotation

Click the right button and select the **Translate** option. Move the centre of the coil to a location in the middle of the 3d Viewer window (again by holding down the left mouse button and dragging the mouse). Figure 8.13 shows a screenshot after the translation.

Next click the right button and select the **Zoom** option. Enlarge the model by dragging with the left mouse button pressed. Figure 8.14 shows a screenshot after zooming in.

This view can also be copied to the Graphics Window by selecting

DISPLAY ↓
Display Command... ...copy 3d view

Figure 8.15 shows the results.

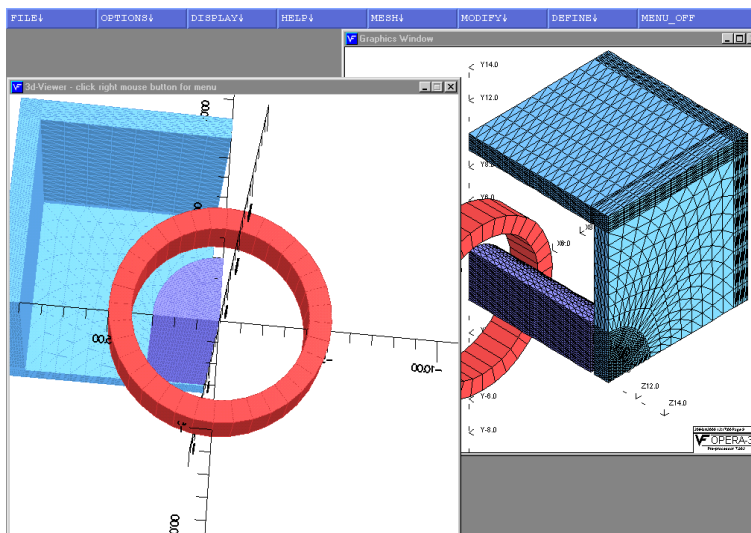


Figure 8.13 3d Viewer after translation

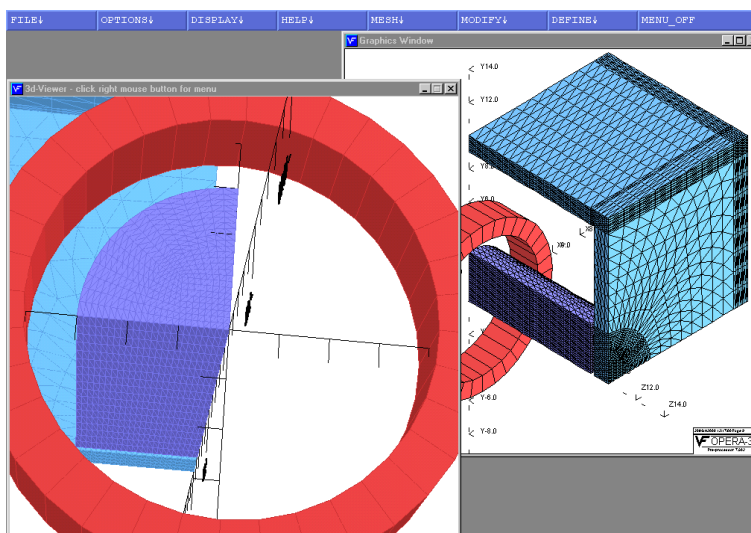


Figure 8.14 3d Viewer after zooming in

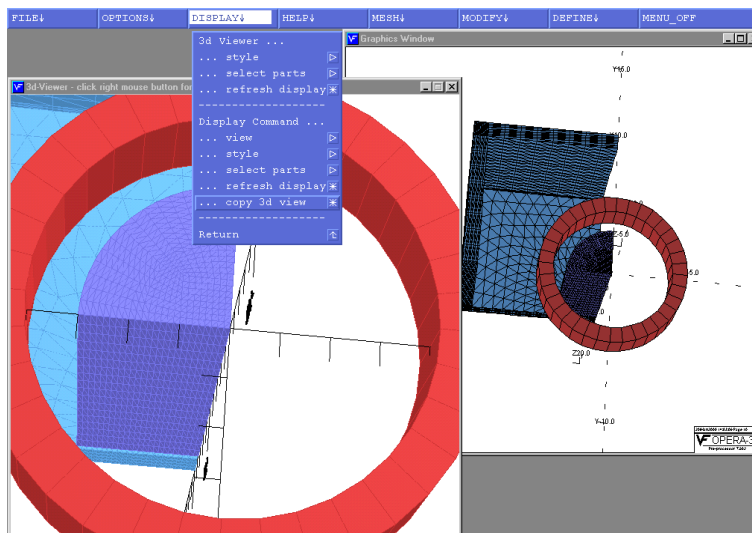


Figure 8.15 Screenshot after copying the 3d view to the graphics window

Saving TOSCA Data

The instructions issued to create the model should be saved so that the pre processing may be repeated or modified as required. If this is done *after* creating the analysis data, the analysis options will be recorded as well.

Creating the TOSCA database file

If you have the TOSCA analysis module installed and wish to solve the solenoid model as a static solution, select

```
FILE ↓
  Analysis
    ... create new database → Statics (TOSCA)
                           Magnetostatics
```

and select the units and element type to be used by completing the parameter box as shown below:

New Analysis Database File

File

Units: ☒ CGS ☐ SI (metres) ☐ SI (mm) ☐ SI (Microns) ☐ SI (Inches)

Element type: ☒ Linear ☐ Quadratic ☐ Mixed

followed by **Accept**. If the box ↓ alongside the file name is selected, a file selection box is raised, allowing all existing files to be shown/selected.

The element type is selected to be linear. This forces all elements in the model to have a linear interpolation in the solver. If the element type is set to mixed, the linear/quadratic information, which was given through the material definition stage, will be used. If the element type is set to quadratic, all elements in the model will be quadratic, which will give a greater accuracy than using linear elements. However the overall number of equations will increase, so this option must be used with care.

It is recommended that quadratic hexahedra are used wherever possible, to ensure the greatest accuracy. In this example linear elements are used for reasons of size of the database. As this model will be examined later with ELEKTRA, every node in a **VECTOR** potential region will have 4 equations instead of one.

After a short pause whilst the main database is prepared, a further parameter box is presented, allowing a title to be specified for the model. Any number of lines of text can be input, and the title is terminated by typing a single * character on a line on its own.

Having cleared the message box, the **TOSCA (Magnetic)** data menu appears

TOSCA (Magnetic)	
Materials	→
Linear solution	<input checked="" type="checkbox"/>
Non-linear solution	<input type="checkbox"/>
Adaptive RHS Integrals	<input type="checkbox"/>
Periodicity conditions	→
External fields	→
Add drive fields	→
Automatic potential cuts	<input type="checkbox"/>
<hr/>	
Check data	*
<hr/>	
Return	←

The material properties of the different media in the model are first defined, by selecting **Materials**. For each material in the model, the linear or non-linear characteristics can be defined.

In the present model, a dialog appears:

The dialog box is titled "Material Names". It contains a table with two columns: "Material Names" and "Properties of air". The table has two rows: "CORE" and "BOX". The "CORE" row is highlighted. Below the table, there are checkboxes for "Material Characteristics": "Linear" (checked), "Non-linear" (unchecked), "Isotropic" (checked), "Packed" (unchecked), and "Multiple" (unchecked). At the bottom, there are two buttons: "Define" and "Return".

Material Names	Properties of air
CORE	Linear, isotropic
BOX	

Material Characteristics

☒ Linear
 ☐ Non-linear
 ☒ Isotropic
 ☐ Packed
 ☐ Multiple

Selecting **CORE** first, the material properties can be defined. These are to be linear isotropic. Selecting **Define** allows the properties of the material for **CORE** to be defined. The parameter box that appears should be completed to set an isotropic permeability of 300, and coercive force remaining zero. Follow this by selecting **Accept**.

The same is done for material **BOX**, by highlighting the button alongside **BOX**, and selecting **Define**. Again, the isotropic permeability is set to 500, with zero coercive force, and select **Accept**. Close the material definition box by selecting **Return**.

We leave all other features in the **TOSCA (Magnetic)** data menu without changing; so we will perform a linear solution, and there are no external fields etc.

Select **Return** in the **TOSCA (Magnetic)** data menu, and check that the data is correct in the information box. Close the information box, to complete storing the analysis database.

```
File: inductor.op3 simulation:      1
Created on: 03/Oct/2001 14:13:57
In Directory:
... \My Work Folder
By Machine:
Node: Mycomputer. Processor: x86. System: Windows
Log Files: Opera3d_Pre_1.backup/log/lp. Opera-3d V8.5

TOSCA Magnetostatic analysis
User did not enter a title

CGS units

1 conductor (current densities in ACM2)
  1 Solenoid
    Current Densities: 1000.0
Adaptive RHS integrals

Boundary Conditions: NORMMAGN TANGMAGN

Linear solution

2 materials defined in the simulation
CORE  : linear isotropic permeability (300.0)
BOX   : linear isotropic permeability (500.0)

17425 nodes in the model
Only linear elements exist within the model
15360 linear hexahedra
```



Continue

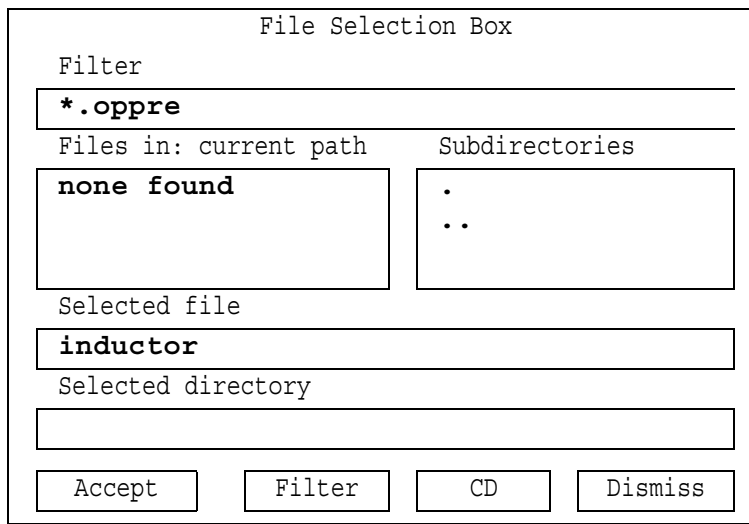
Writing the Pre Processor Data File

To save the pre processor commands, select

FILE ↓

Write pre-processor file

This brings up a File Selection Box.



The image shows a 'File Selection Box' dialog box. It has a title bar 'File Selection Box'. Inside, there is a 'Filter' section with a text box containing '*.oppre'. Below this, there are two columns: 'Files in: current path' and 'Subdirectories'. The 'Files in: current path' column contains the text 'none found'. The 'Subdirectories' column contains the text '.' and '..'. Below these columns, there is a 'Selected file' section with a text box containing 'inductor'. Below that is a 'Selected directory' section with an empty text box. At the bottom, there are four buttons: 'Accept', 'Filter', 'CD', and 'Dismiss'.

The desired filename can be included (for example *inductor*) in the box labelled “Selected File”. Finish by selecting **Accept**.

Leaving the Pre Processor

The data has now been stored correctly. Exit the pre processor so that the analysis module may be used. Do this by selecting

FILE ↓
 End OPERA-3d/Pre

and confirm with **YES**.

Running TOSCA On-Line

UNIX Operating Systems

From within the OPERA-3d environment select the TOSCA option

Option:

tosca

and from the following prompt:

Please give TOSCA database filename (without the .OP3 suffix)

inductor

and choose that the analysis is carried out immediately:

Do you want to run the analysis now or later? (n or l)

n

The analysis then proceeds automatically.

Windows Systems

Choose interactive solution under the OPERA-3d menu in the OPERA Console. Select the TOSCA solver and browse to where the *inductor.op3* file was saved and select it. The analysis module will then proceed with the calculations.

Entering OPERA-3d Post Processor

UNIX Operating Systems

From the OPERA-3d environment enter the post processor by selecting:

Option:

post

Windows Systems

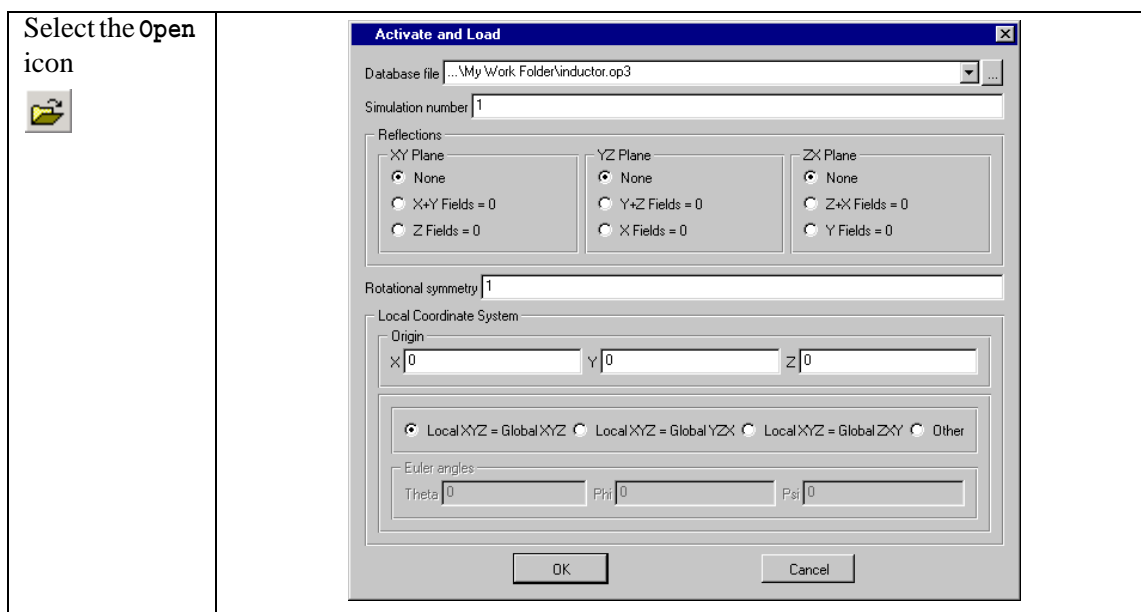
Launch the 3D post processor from the OPERA Console. The OPERA Console can be started with the supplied system tray icon.

Examining the Basic Solution from TOSCA

The modelling procedure has made use of the symmetry of the device. The basic solution is without reflection or rotation symmetries shown.

Loading the Model in the Post Processor


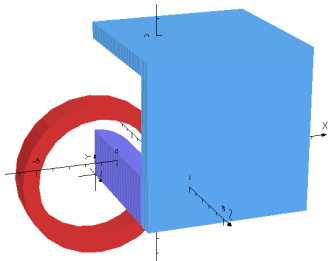

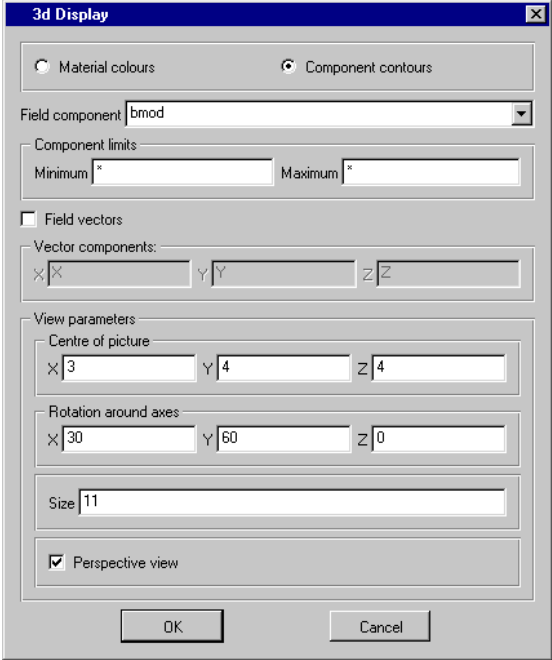
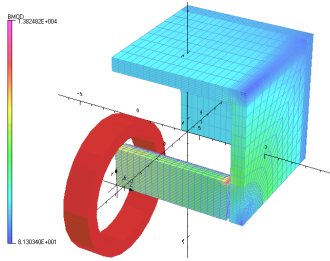
First the solution file needs to be activated and loaded.




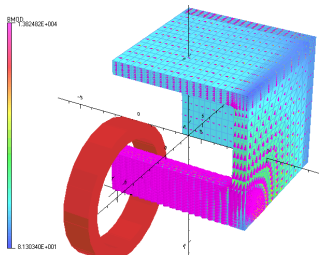
Enter the filename *inductor.op3* or use the ... button to browse directories and locate the file.

Viewing the Model

Once the file has been activated a display of the flux density on the surface of the model can be examined.


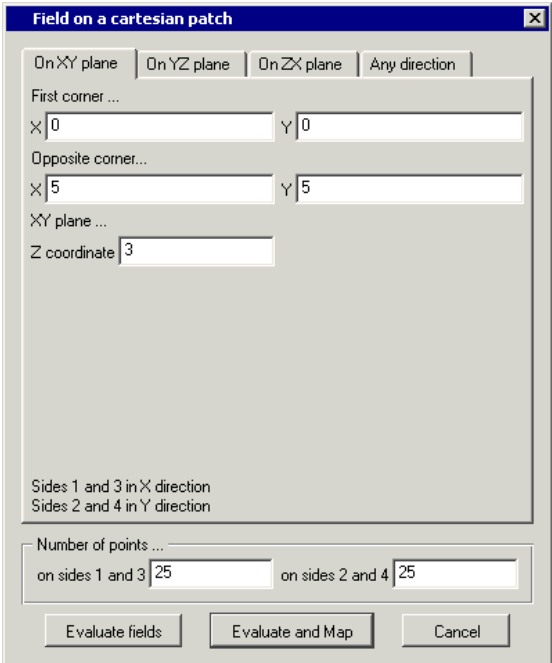
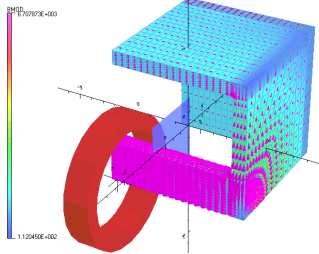


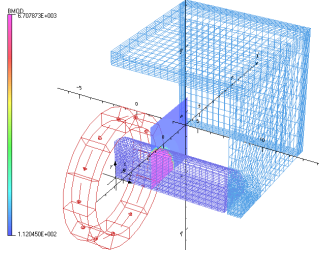
<p>SelectDefault select and refresh icon</p> 	<p>This will select all materials not called air for display by default. The timer bar shows how the selection and display process is proceeding.</p>	
<p>Select 3D display icon</p> 		


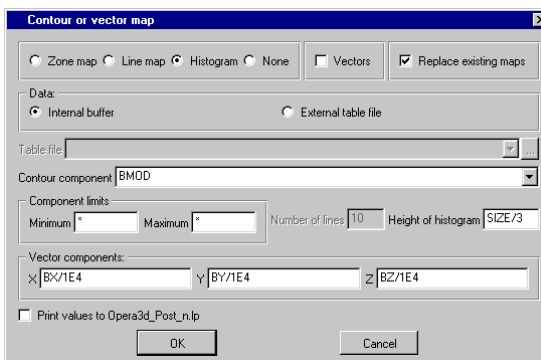
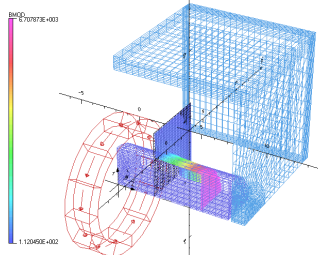
In addition to viewing the field contours it is possible to display field vectors on the surface elements as well.

<p>Select 3D display icon</p> 	<p>The vectors are scaled down by 1E4 so that they are a sensible size to fit the display.</p> <div data-bbox="459 499 1015 1157"><p>3d Display</p><p><input type="radio"/> Material colours <input checked="" type="radio"/> Component contours</p><p>Field component: bmod</p><p>Component limits Minimum: Maximum: </p><p><input checked="" type="checkbox"/> Field vectors</p><p>Vector components: X: bx/1e4 Y: by/1e4 Z: bz/1e4</p><p>View parameters Centre of picture X: 3 Y: 4 Z: 4</p><p>Rotation around axes X: 30 Y: 60 Z: 0</p><p>Size: 11</p><p><input checked="" type="checkbox"/> Perspective view</p><p>OK Cancel</p></div>	
-----------------------------------------------------------------------------------------------------------------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-------------------------------------------------------------------------------------

2D surfaces in the 3D model space

To examine a 2D surface in the model, it is possible to specify a surface or “patch”.


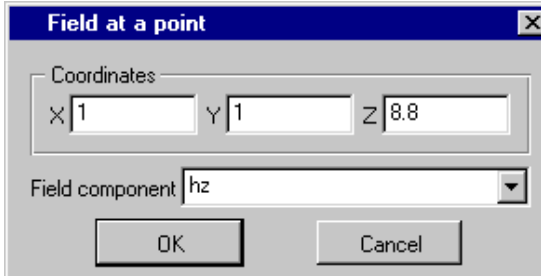
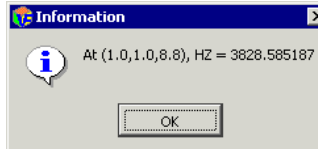

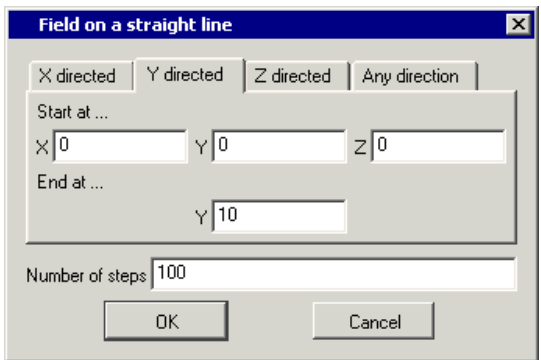
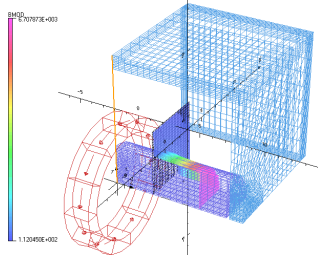
<p>Select the Fields on a Cartesian patch icon</p> 	<p>Select the On XY Plane tab, complete the dialog, and select Evaluate and map.</p> 	
<p>Toggle the Solid view of model icon</p>  <p>Toggle the Vectors on surface icon</p> 	<p>This can be better seen by turning off the surface elements in order to see a wire frame view of the model. Also turn off the display of the vectors.</p>	

<p>Select the Con- tour or vec- tor map icon</p> 	<p>Alternatively the map can be displayed as a histogram over the model.</p> 	
---------------------------------------------------------------------------------------------------------------------------------------------------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------	-------------------------------------------------------------------------------------

Evaluation of solution at points and along lines

It is possible to examine the solution at points and lines in the finite element mesh.

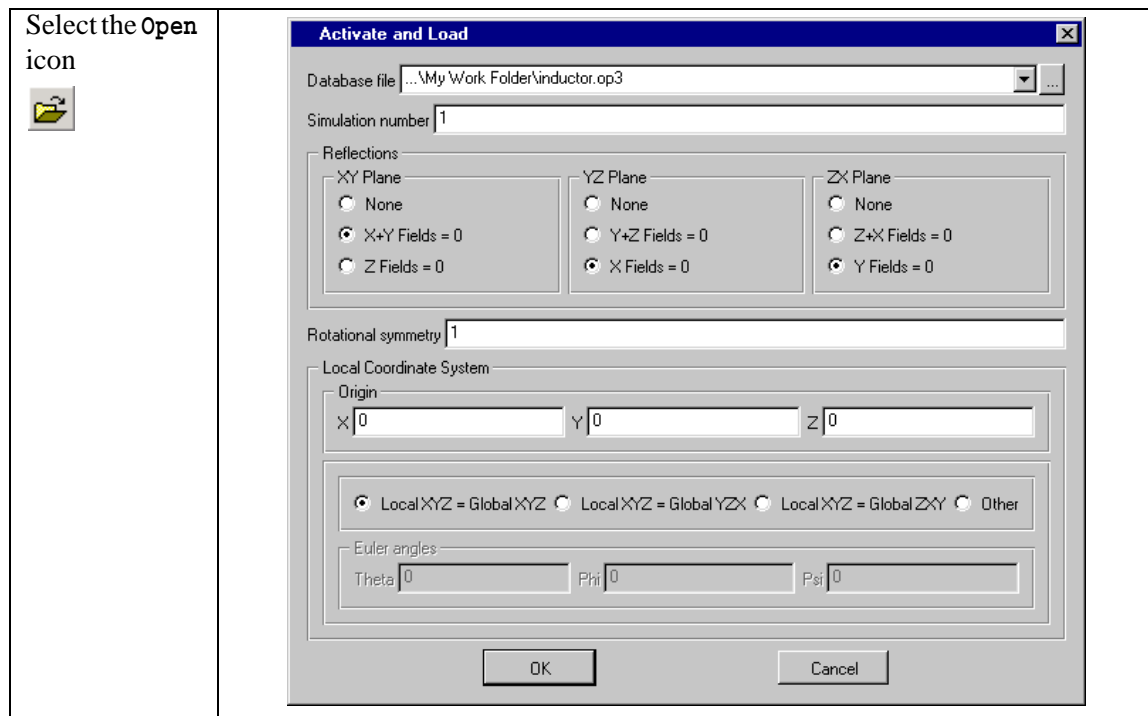
To examine the value of the magnetic field in the z-direction at a point and along a line the following actions can be taken.

<p>Select the Fields at a point icon</p> 		
<p>Select the Fields on a straight line icon</p> 		

Examining the complete device from TOSCA

Loading the model

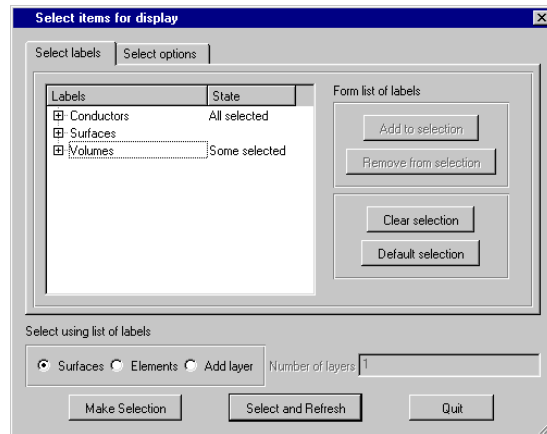
To examine the results of the complete device, the solution should be loaded with all the required reflections and rotations. Select **File** → **Clear all data** to reset the default settings.



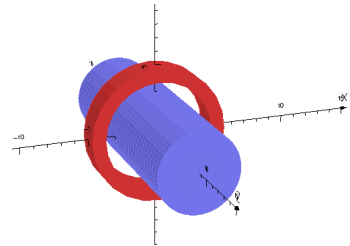
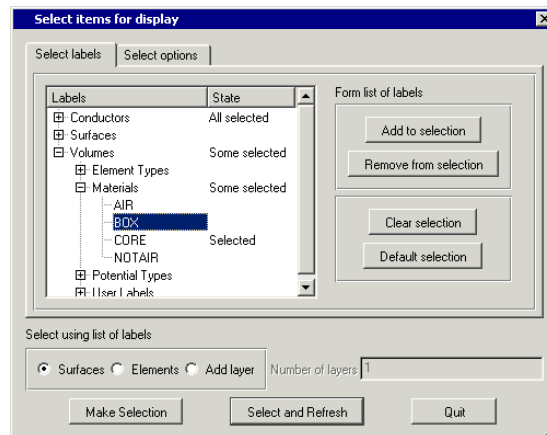
Select the
Select icon

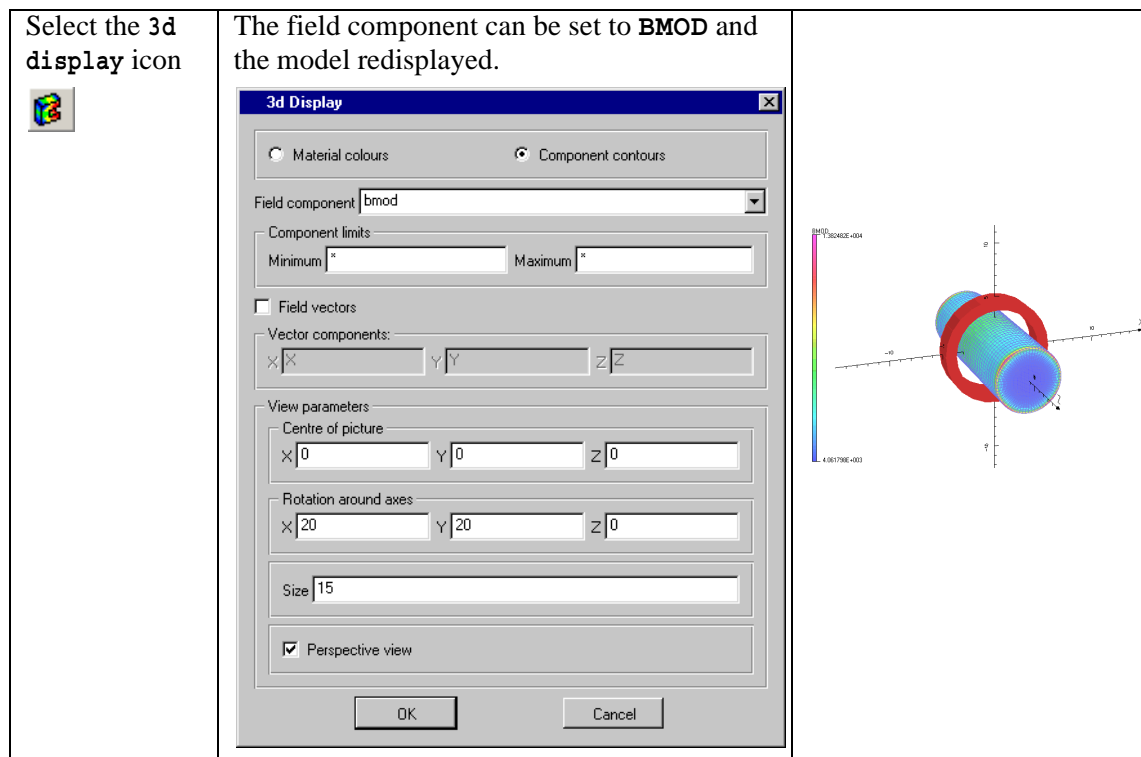


Select **Default** selection first.



Select the **BOX** material and then choose **Remove from selection** so that only the **CORE** is selected for display when the **Select and Refresh** button is pushed.





This completes the post processing at this stage and the post processor can be closed by selecting

File → Exit

ELEKTRA Worked Example

The data prepared in the previous model can also be used as the basis of an ELEKTRA analysis, calculating the eddy currents in the non-air volumes.

In this chapter, the data prepared in the previous chapter will be suitably modified, and analysed with ELEKTRA.

Reading the Existing TOSCA Model Data

After starting the pre processor in the usual way, the first action is to read the previously defined TOSCA model data, already prepared. This is done by selecting:

FILE ↓

Read pre-processor file

A file selection box appears, from which the previous pre processor file can be selected (for example *inductor.oppre*). Either double-click on the file name, or single-click, and hit **Accept**.

The screenshot shows a 'File Selection Box' dialog. At the top is a 'Filter' field containing '*.oppre'. Below this are two panes: 'Files in: current path' which lists 'inductor.oppre', and 'Sub-directories' which lists '.' and '..'. Below the panes are two empty text fields labeled 'Selected file' and 'Selected directory'. At the bottom are four buttons: 'Accept', 'Filter', 'CD', and 'Dismiss'.

Clear the message box that appears when the pre processor file has finished loading.

Additional Modelling for ELEKTRA

If the ELEKTRA analysis module is to be used, the eddy current effects must be modelled with an additional type of potential, that is **VECTOR POTENTIAL**.

The database has already been defined with **VECTOR** potentials, although with the TOSCA analysis they were interpreted as **TOTAL** scalar potentials. No changes are needed to the model therefore, and so a new simulation can be added to the existing database.

Writing the ELEKTRA database file

If you have the ELEKTRA analysis module installed and wish to solve the inductor model as an ac solution, select

FILE ↓

Use existing database

and under the section **Low Frequency (ELEKTRA)**, select **Steady-state AC**. The following parameter box is then completed, using the previous file name:

Existing Analysis Database File

File ↓

☐ Add new case
(New coils, boundary conditions, materials, options)

☒ Copy from case
(New materials and analysis options)

☐ Edit pending case

☐ Restart case

Case number

Accept Quit

Warning
Edit, Copy and Restart
might corrupt the data
in the pre processor

If the box ↓ alongside the file name is selected, a file selection box is raised, allowing all existing files to be shown/selected.

Select **Accept**, clear the message box that appears.

The first action is to give a title for this simulation, terminated with an * character on a line of its own. A prompt is then given for the drive phase. Since this is an ELEKTRA-SS solution, it is possible to set the phase of the drive (assumed to be

Cosine). There is only a single drive, so in this case set the phase to zero, and **Return**.

Having cleared the message box, the **ELEKTRA (steady state)** data menu appears. Now it is necessary to specify the analysis specific quantities.

ELEKTRA (steady state)	
Materials	→
Linear solution	<input checked="" type="checkbox"/>
Non-linear solution	<input type="checkbox"/>
Adaptive RHS Integrals	<input type="checkbox"/>
Add drive fields	→
Drive frequencies	→
Check data	*
Return	←

First select **Materials**. For each material in the model, the linear or non-linear characteristics can be defined. For example, in the present model, the parameter box will appear, and **CORE** is selected as follows:

Material names	
CORE	Linear isotropic
BOX	Linear isotropic

Material Characteristics		
<input checked="" type="checkbox"/> Linear	<input type="checkbox"/> Non-linear	
<input checked="" type="checkbox"/> Isotropic	<input type="checkbox"/> Packed	<input type="checkbox"/> Multiple
Define		Return

Selecting **CORE** first, the material properties can be defined. These are to be linear isotropic. Selecting **Define** allows the properties of the material for **CORE** to be defined.

The parameter box that appears should be completed to set an isotropic permeability should be 1 (previously set at 300). The isotropic conductivity should be set to 1.0e5 (with zero complex phase lag). Follow this by selecting **Accept**.

Properties for material CORE

Linear Permeability

Isotropic

Complex phase lag

Linear conductivity [SCM]

Isotropic

Complex phase lag

The same is done for material **BOX**, by highlighting **BOX**, and selecting **Define**. Again, the isotropic permeability should be changed to 1 (from 500), with conductivity of 1.0e4, and select **Accept**.

Close the **Material names** box by selecting **Return**.

The frequency of the solution must be given as well. So select **Drive frequencies** and enter **50** for the new frequency. Press **Add** to add this frequency to the case list, and close the menu with **Return**.

Close the **ELEKTRA (steady state)** data menu with **Return**, and check that the data is correct in the information box. This should be:

```

File: inductor.op3 simulation:      2
Created on: 19/Nov/2001 16:41:46
In Directory:
...\My Work Folder
By Machine:
Node: Mycomputer. Processor: x86. System: Windows
Log Files: Opera3d_Pre_1.backup/log/lp. Opera-3d V8.5

ELEKTRA Steady state harmonic analysis,
User did not enter a title

CGS units

1 conductor (current densities in ACM2);
    1 solenoid
    current densities: 1000.0
Adaptive RHS integrals
Linear solution

Drive sets and functions
ONE  : Coil drive type Cosine,
      (Frequency by case, Phase 0.0)

Boundary Conditions: NORMMAGN TANGMAGN

Linear solution

2 materials defined in the simulation
CORE : linear isotropic conductivity (100000.0 SCM)
BOX  : linear isotropic conductivity (10000.0 SCM)

1 solution frequency defined: 50.0

17425 nodes in the model
Only linear elements exist within the model
15360 linear hexahedra

```



Continue

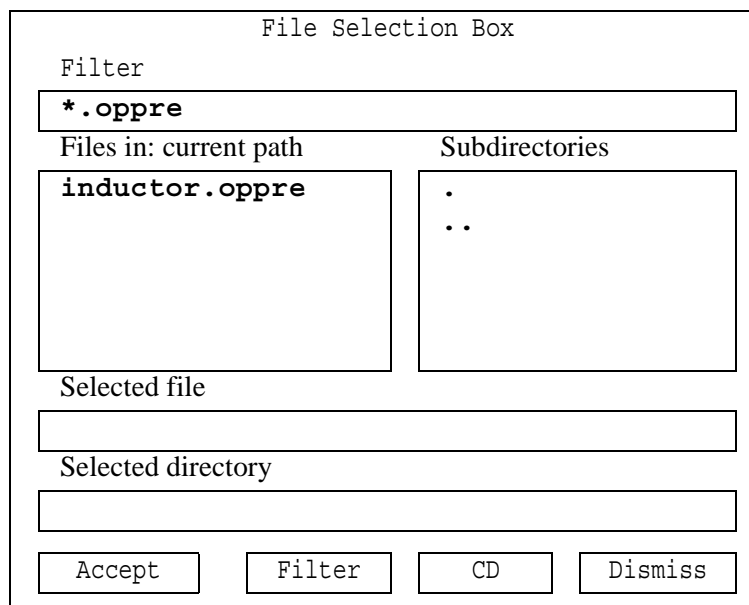
Close the information box, to complete storing the analysis database.

Writing the Pre Processor Data File

To save the pre processor commands, select

FILE ↓
 Write pre-processor file

This brings up a File Selection Box.



The File Selection Box dialog is titled "File Selection Box". It contains a "Filter" field with the text "*.oppre". Below the filter, there are two columns: "Files in: current path" and "Subdirectories". The "Files in: current path" column contains the file "inductor.oppre". The "Subdirectories" column contains "." and "..". Below these columns, there are two empty text boxes labeled "Selected file" and "Selected directory". At the bottom, there are four buttons: "Accept", "Filter", "CD", and "Dismiss".

into which the filename can be included. The existing file can be updated by double-clicking this filename, and allowing it to be over-written.

Leaving OPERA-3d Pre Processor

The data has now been stored correctly. Exit the pre processor so that the analysis module may be used. Do this by selecting

FILE ↓
 End OPERA-3d/Pre

and confirm with **YES**.

Running ELEKTRA On-Line

UNIX Operating Systems

From within the OPERA-3d environment select the ELEKTRA option

Option:

elektra

and from the following prompt:

Please give ELEKTRA database filename (without the .OP3 suffix)

inductor

Indicate which type of ELEKTRA solution is required (SS in this case)

Steady state AC (ss), Transient (tr) or Velocity (vl)?

ss

and choose that the analysis is carried out immediately:

Do you want to run the analysis now or later? (n or l)

n

The analysis then proceeds automatically.

Windows Systems

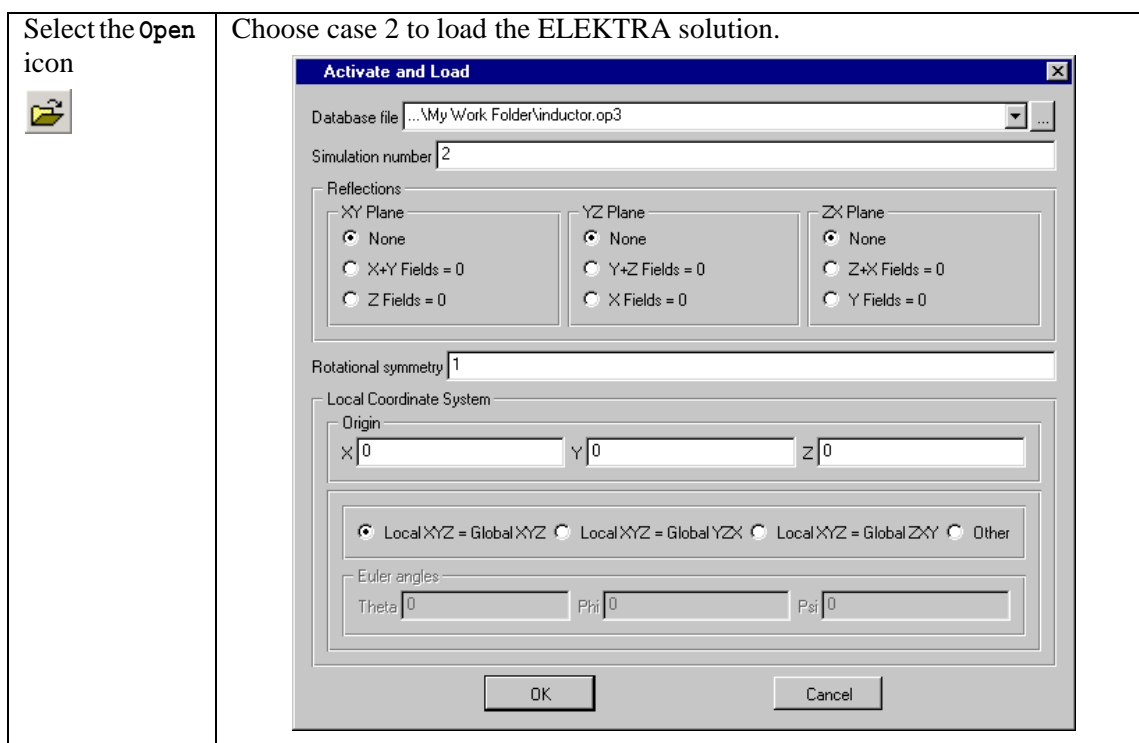
Choose interactive solution under the OPERA-3d menu in the OPERA Console. Select the ELEKTRA-SS solver and browse to where the *inductor.op3* file was saved and select it. The analysis module will then proceed with the calculations.


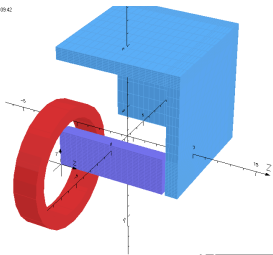

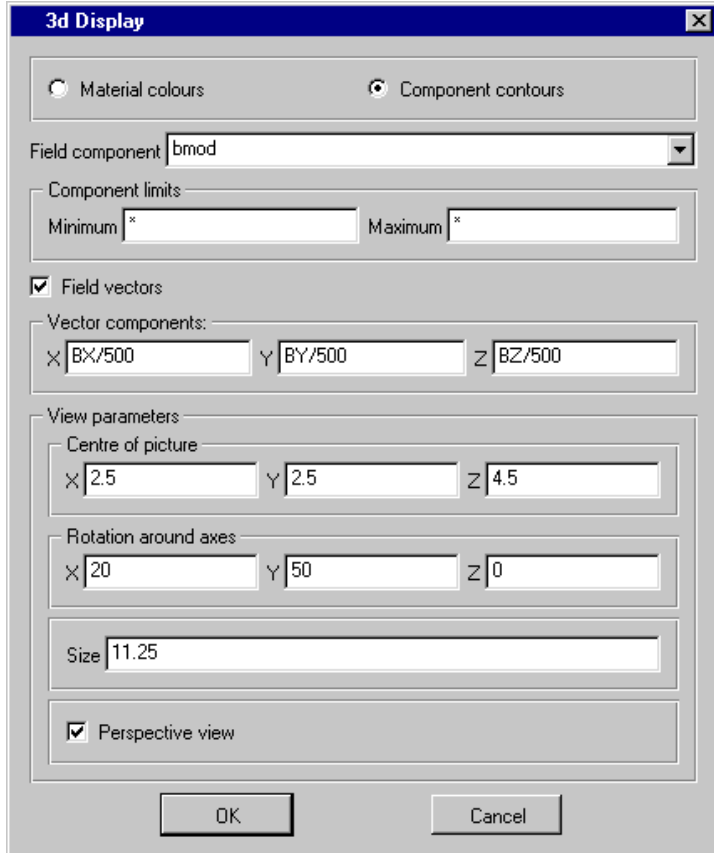
Examining the Basic Solution from ELEKTRA

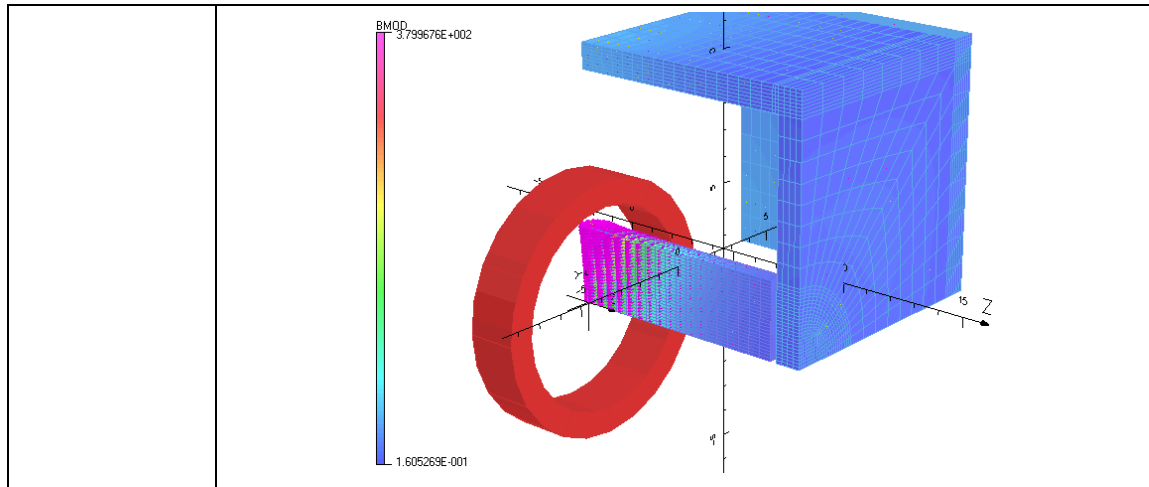
All the commands and techniques used in the post processing of the TOSCA solution may be used to examine the ELEKTRA solution once the model has been read into the post processor.

Loading and Displaying the Model in the Post Processor

The new solution is read in and displayed in the same way as in the earlier TOSCA solution, except that as the permeability of the **CORE** and **BOX** have both now been changed to 1. As a result the magnetic flux density will be much smaller and so the vector display of the flux density has to be rescaled to a value of $B_x/100$, $B_y/100$, $B_z/100$ in order to see the vectors.



<p>SelectDefault select and refresh icon</p> 	<p>This will select all materials not called air for display by default. The progress bar shows how the selection and display process is proceeding.</p>	<p>11/04/2001 10:59:42</p>  <p>VECTOR FIELDS</p>
<p>Select the 3d display icon</p> 	<p>The vectors are scaled by 500 so that they are a sensible size to fit the display.</p>	




The other post processing commands that were used for the TOSCA solution can now be used on the ELEKTRA solution (noting that H_z is not available, so replace it in the script with B_z)

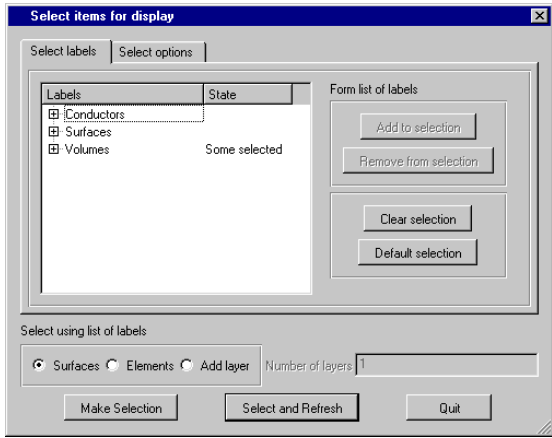
Eddy currents in the ELEKTRA solution

In addition, the eddy current effects can be seen. In this case, however we will now display the model with the conductor hidden from view.

Select the **Select items for display** icon


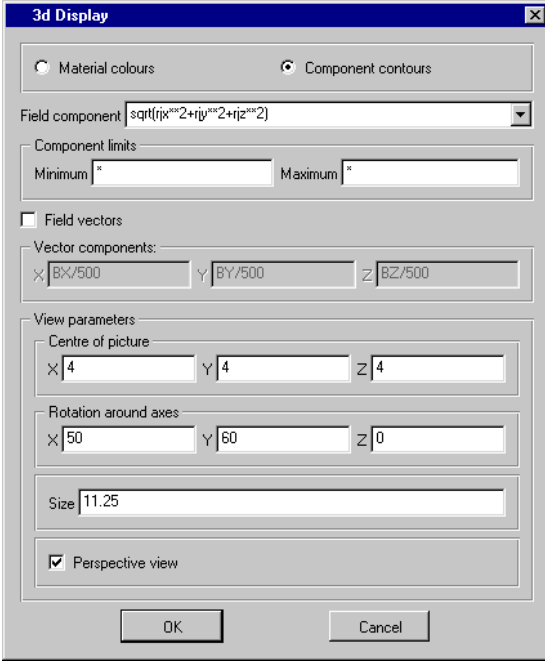
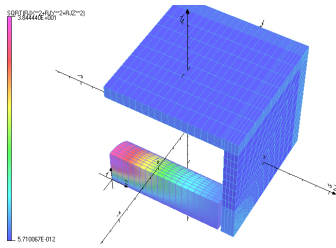

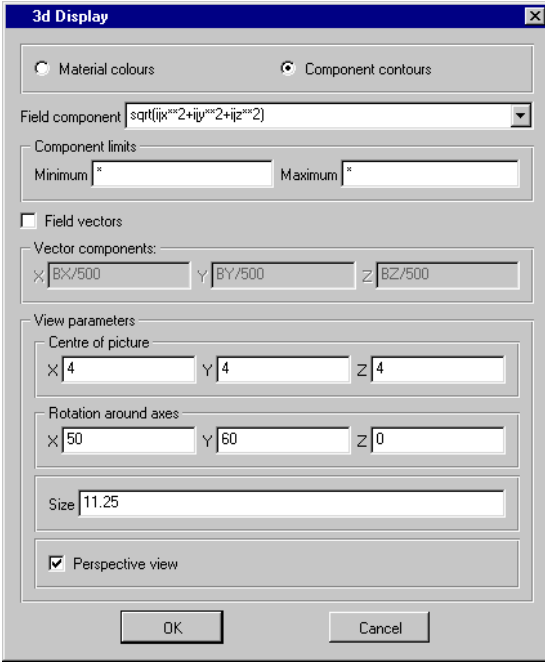
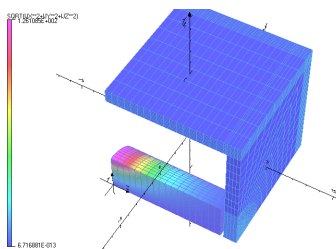


The following dialog can be used to select only certain parts of the model for display and calculation.


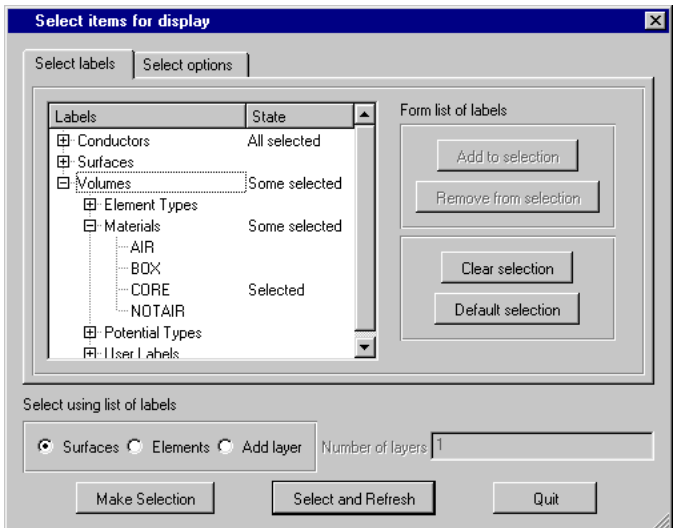


Select the item **Conductors**, and use the **Remove from selection** button to remove the conductors from the display list. Apply the selection using the **Make**

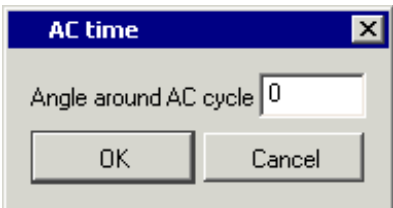
Selection button. Now the display can be changed to show the in-phase component of the complex ELEKTRA solution.

<p>Select the 3d display icon</p> 	 <p>The dialog box shows the following settings:</p> <ul style="list-style-type: none"> Material colours: <input type="radio"/>; Component contours: <input checked="" type="radio"/> Field component: $\sqrt{(jx^{**2}+iy^{**2}+jz^{**2})}$ Component limits: Minimum: "", Maximum: "" Field vectors: <input type="checkbox"/> Vector components: X: $Bx/500$, Y: $By/500$, Z: $Bz/500$ View parameters: <ul style="list-style-type: none"> Centre of picture: X: 4, Y: 4, Z: 4 Rotation around axes: X: 50, Y: 60, Z: 0 Size: 11.25 Perspective view: <input checked="" type="checkbox"/> 	 <p>A 3D plot showing the in-phase component of the complex ELEKTRA solution. The plot is a blue cube with a color bar on the left ranging from 0 to 5.71000E-012.</p>
<p>Select the 3d display icon</p> 	<p>The quadrature component can also be displayed.</p>  <p>The dialog box shows the following settings:</p> <ul style="list-style-type: none"> Material colours: <input type="radio"/>; Component contours: <input checked="" type="radio"/> Field component: $\sqrt{(jx^{**2}+iy^{**2}+jz^{**2})}$ Component limits: Minimum: "", Maximum: "" Field vectors: <input type="checkbox"/> Vector components: X: $Bx/500$, Y: $By/500$, Z: $Bz/500$ View parameters: <ul style="list-style-type: none"> Centre of picture: X: 4, Y: 4, Z: 4 Rotation around axes: X: 50, Y: 60, Z: 0 Size: 11.25 Perspective view: <input checked="" type="checkbox"/> 	 <p>A 3D plot showing the quadrature component of the complex ELEKTRA solution. The plot is a blue cube with a color bar on the left ranging from 0 to 5.71000E-013.</p>

Before another method of displaying the in-phase part of the current density is shown, we switch off the surrounding box and add the coil.

<p>Select the Select icon</p> 	<p>Use the Remove from selection button to remove material BOX from the display list. Use the Add to selection button to include the Conductor. Select the Make Selection button.</p> 
------------------------------------------------------------------------------------------------------------------------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

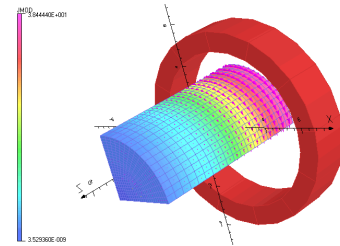
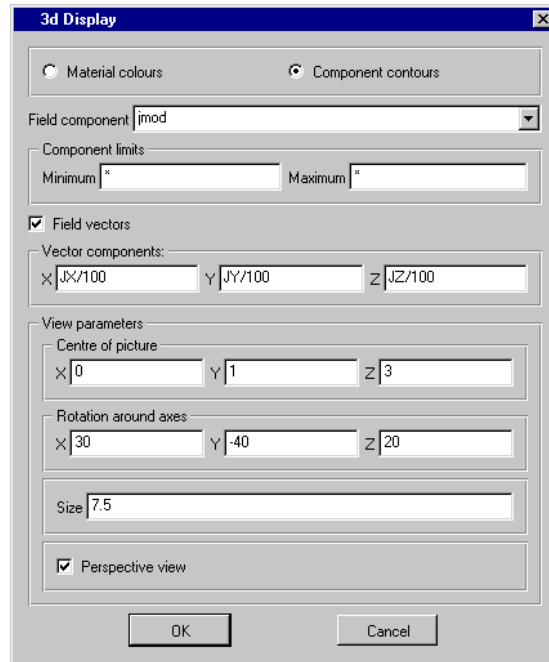
To display the current density on the surface of the model at a particular time on the AC cycle a value of the AC time can be entered.

Options → AC time		
	<p>Accept the default value of 0 the in-phase current density can be displayed.</p> 	

Select the **3d display** icon



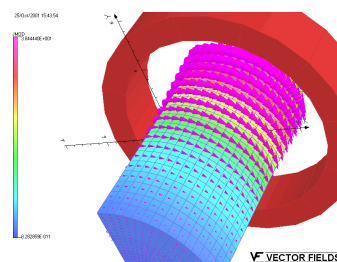
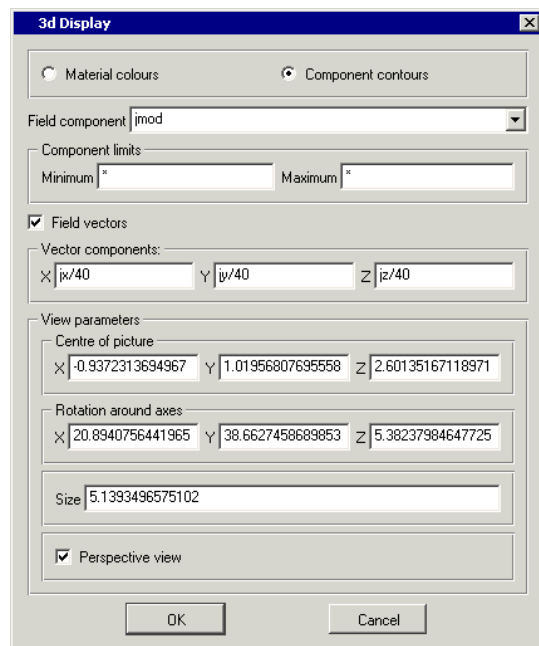
Changing the field component to **JMOD** updates the display for the central pole. Also include vector of current density.



Select the **3d display** icon



As the current density is a vector quantity the direction can be shown by adding vectors to the display.



Leaving OPERA-3d Post Processor

Exit the post processor so that other options are available. Do this by selecting

File ↓

Exit

and confirm with **YES**

Chapter 9

Speaker Analysis Using TOSCA

Introduction

This tutorial illustrates how to model a speaker core using the pre and post processor within OPERA-3d, together with the analysis module TOSCA.

This model will be built using the automatic mesh generation facility, giving a tetrahedral mesh throughout the model. The model could also be built in terms of hexahedra, as in the previous model, without much extra work involved. However, in order to show more features, tetrahedra will be used.

The first section is a step by step guide to the pre processing, i.e. the building of the model. This model is then passed to the TOSCA analysis module.

The pre processor data file (*.oppre*) is also provided within the OPERA installation (see the sub-folder *Examples/3D*).

The second section is a guide to the use of the post processor. Typical commands are used to illustrate methods of examining the results.

Figure 9.1 shows a loudspeaker to be modelled using TOSCA. The symmetry of the model means that only 1/8th of the full speaker needs to be modelled. Figure 9.2 shows the FE-model of the speaker. The components can be identified, consisting of a cylindrical piece of a rare earth magnet inset within an iron yoke. The field generated is due to the permanent magnet, and a coil placed within the air gap in the yoke.

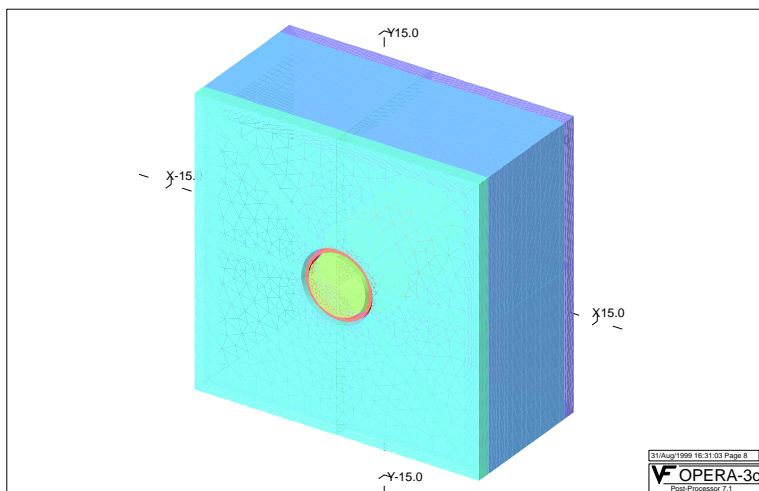


Figure 9.1 The complete speaker

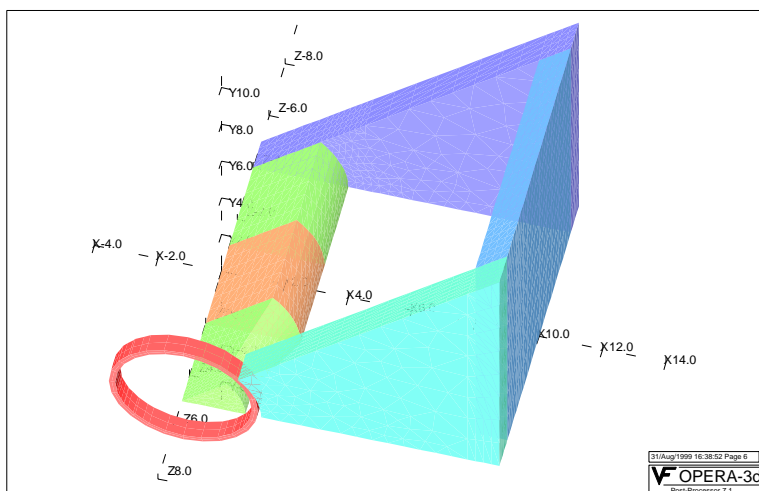


Figure 9.2 The section of the speaker to be modelled in OPERA-3d

Pre Processor

Launch the OPERA-3d pre processor in the usual way (see “Starting OPERA-3d Pre Processor” on page 8-1).

The following pre processing steps will be undertaken.

1. The definition of the baseplane, including: creating construction lines, creating points to be used within the model, creating facets and setting the subdivisions.
2. Extrusion of this baseplane to create the full 3 dimensional model.
3. Modification to the model to adjust the shape, material properties and boundary conditions for the model.
4. Definition of the speaker coil within the air gap.
5. Generation of the file to be sent to the TOSCA analysis module.

Creating the Baseplane

Creating a baseplane is the first step towards building a new mesh. Select

DEFINE ↓

Define new mesh → Finite element mesh → XY plane,
extrude in Z

The baseplane will be placed at a coordinate of -50 . This is because the first extrusion layers will represent air regions surrounding the model. After extrusions, the key regions in the model will lie centred on the (0,0,0) position. Complete the dialog box as shown below and press **Return**

W coordinate of base plane	=	- 50
----------------------------	---	------

Initially set the viewing coordinates within the baseplane to

Minimum on horizontal axis	=	0
Maximum on horizontal axis	=	10
Minimum on vertical axis	=	0
Maximum on vertical axis	=	10
<div style="display: flex; justify-content: space-around; width: 100%;"> <div style="border: 1px solid black; padding: 5px 20px;">Accept</div> <div style="border: 1px solid black; padding: 5px 20px;">Dismiss</div> </div>		

Press **Accept** to accept the default model axis.

Point Definition Menu

The menu system now displays the **Define Baseplane** menu. Select **Point Input**. From this **Point Definition** menu it is possible to define points that will be used within the baseplane. Points can be created by specifying coordinates directly. In this case however, construction lines will be generated. These are either straight lines or arcs. Using the intersections of these lines to set the point coordinates provides a simpler method of positioning points without direct calculation of coordinates.

To create the straight construction lines, a start and end coordinate must be given. Lines can also be rotated by an angle about the start point. Select

... **Construction lines** → **Enter C_Lines** → **By parameters** → **Line**

Start.	U	=	0
.....	V	=	0
Finish	U	=	50
.....	V	=	0
Rotation		=	0
Accept		Dismiss	

and repeat this for the following straight construction lines

Start.	U	=	0
.....	V	=	0
Finish	U	=	50
.....	V	=	0
Rotation		=	22.5
Accept		Dismiss	

Start.	U	=	0
.....	V	=	0
Finish	U	=	50
.....	V	=	0
Rotation		=	45
Accept		Dismiss	

Start.	U	=	9
.....	V	=	0
Finish	U	=	9
.....	V	=	10
Rotation		=	0
<input type="button" value="Accept"/>		<input type="button" value="Dismiss"/>	

Start.	U	=	10
.....	V	=	0
Finish	U	=	10
.....	V	=	10
Rotation		=	0
<input type="button" value="Accept"/>		<input type="button" value="Dismiss"/>	

Start.	U	=	15
.....	V	=	0
Finish	U	=	15
.....	V	=	15
Rotation		=	0
<input type="button" value="Accept"/>		<input type="button" value="Dismiss"/>	

Further from the origin, points are needed to help model the surrounding air. First resize the display by choosing **Return** 3 times to return to the point definition menu, and selecting

... Redraw Picture → Numerical limits

Minimum on horizontal axis	=	0
Maximum on horizontal axis	=	50
Minimum on vertical axis	=	0
Maximum on vertical axis	=	50
<input type="button" value="Accept"/>		<input type="button" value="Dismiss"/>

... Construction lines → Enter C_Lines → By parameters → Line

Start.	U	=	50
.....	V	=	0
Finish	U	=	50
.....	V	=	50
Rotation		=	0
Accept		Dismiss	

Also needed are some arc construction lines. These are defined by a centre point, a start radius and angle, and an end radius and angle. Change the display back to the previous size by using **Return** (to the point definition menu) and then resize the screen to its previous limits by

... Redraw Picture → Previous size

Now select

... Construction lines → Enter C_Lines → By parameters → Arc

Centre U	=	0
..... V	=	0
Start R	=	3
...Theta	=	0
End R	=	3
...Theta	=	45
Accept		Dismiss

and repeat to generate the following arc construction lines

Centre U	=	0
..... V	=	0
Start R	=	4
...Theta	=	0
End R	=	4
...Theta	=	45
Accept		Dismiss

Centre U	=	0
..... V	=	0
Start R	=	5
...Theta	=	0
End R	=	5
...Theta	=	45
Accept		Dismiss

All the construction lines have now been created. To generate points at the intersection of these lines, select **Return** 3 times to return to the **Point definition** menu and then choose

... At C_line Intersection

and click the mouse near to the point

0, 0

A cross will appear at the intersection of the construction lines to mark the point (after every mouse selection).

Now the following points can be defined¹:

3, 0
 4, 0
 5, 0
 9, 0
 10, 0
 15, 0
 2.1, 2.1
 2.9, 2.9
 3.6, 3.6
 9, 9
 10, 10
 2.8, 1
 3.7, 1.5

The remaining points can only be selected after resizing the display. This is done by selecting:

... Redraw Picture → Numerical limits

1. NOTE: Some points may be hidden by the position of the menus. The function key **F1** acts as a toggle. Pressing the **F1** key will hide the menus. Pressing **F1** again will restore them.

Minimum on horizontal axis	=	0
Maximum on horizontal axis	=	70
Minimum on vertical axis	=	0
Maximum on vertical axis	=	70
<input type="button" value="Accept"/>		<input type="button" value="Dismiss"/>

followed by:

At **C_line Intersection**

Now the following points can be defined:

50, 0
 50, 50
 15, 15

Change the display back to the previous size by

... **Redraw Picture → Previous size**

Close the **Point definition** menu with **Return** and select **Facet Input** from the **Define Baseplane** menu.

Facet Definition Menu

The **Facet Definition** menu is now displayed.

The points that have been created will now be linked to form facets over the base-plane. The facets will be a mixture of 4 sided regions, and general polygonal regions. Curved sections of the model can be formed using *Mid-side* points to create a curved line.

The first facet to be built will represent the circular section of yoke and magnet. From the facet definition menu select **no auto-close** and choose the 3 points

2.1, 2.1
 0, 0
 3, 0

Now select **Mid-side** and click at

2.8, 1

followed by **Close** to force the facet to close with only 3 sides.

The next facet will be used to represent the air gap within the yoke. Select **no auto-close** and select the points

2.9, 2.9
2.1, 2.1

Choose **Mid-side** and click at

2.8, 1

no auto-close

3, 0
4, 0

and finally **Mid-side**

3.7, 1.5

closes the facet.

The next facet will represent the plates at top and bottom of the yoke. Extra nodes are used to improve the mesh discretisation near the edges of the yoke, where the field is likely to be changing most rapidly. The region is defined using the following points¹:

... polygon corner	4, 0	
	5, 0	
	9, 0	
	9, 9	Facet 3
	3.6, 3.6	
	2.9, 2.9	
Mid-side	3.7, 1.5	

and **Close** the polygon.

The next facet will represent the side of the yoke.

Auto-close after 4	9, 0
	10, 0
	10, 10
	9, 9

and the region automatically closes.

One more facet is needed to represent the air outside the yoke. First resize,

1. NOTE: The **F1** key can be used to hide the menus if necessary

Redraw Picture → Numerical limits

Minimum on horizontal axis	=	0
Maximum on horizontal axis	=	70
Minimum on vertical axis	=	0
Maximum on vertical axis	=	70
<input type="button" value="Accept"/>		<input type="button" value="Dismiss"/>

Polygon corner 10, 0
 15, 0
 50, 0
 50, 50 Facet 5
 15, 15
 10, 10

Close

Restore the screen to its previous size.

... Redraw Picture → Previous size

Figure 9.3 shows the facets on the baseplane. This completes the facet definition.

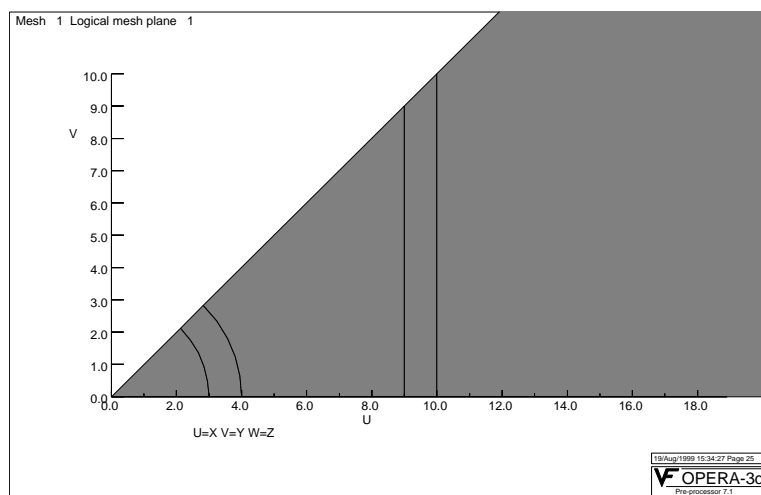


Figure 9.3 The Facets used in the baseplane

Press **Return** to close the **Facet Definition** menu.

Subdivision Menu

The next step is to set the in-plane subdivisions used to create the finite element mesh. Pick **Subdivision** from the **Define Baseplane** menu.

The **Subdivisions** menu is now displayed.

This menu sets the number of divisions along each edge of each facet of the baseplane. This is used when generating the surface and volume element meshes for the analysis program.

Initially a global value of 4 will be selected, i.e. all sides will be set to 4 subdivisions.

Choose the **Set Subdivision** item, select a subdivision of 4 from the box that appears

Subdivision

<input type="checkbox"/> 1	<input type="checkbox"/> 2	<input type="checkbox"/> 3
<input checked="" type="checkbox"/> 4	<input type="checkbox"/> 5	<input type="checkbox"/> 6
<input type="checkbox"/> 7	<input type="checkbox"/> 8	<input type="checkbox"/> 9

Other

Accept

and then **Accept**. From the next menu select **Apply globally**. A message saying that the subdivisions are complete will appear and can be cleared by pressing the **Continue** button. **Return** to close the submenu.

It is now necessary to selectively improve the subdivisions. This is particularly important around the air gap where high accuracy is required, and where the coil fields are changing most rapidly.

Choose the **Set Subdivision**

Subdivision

<input type="checkbox"/> 1	<input type="checkbox"/> 2	<input type="checkbox"/> 3
<input type="checkbox"/> 4	<input type="checkbox"/> 5	<input type="checkbox"/> 6
<input type="checkbox"/> 7	<input checked="" type="checkbox"/> 8	<input type="checkbox"/> 9

Other

Accept

Select **Accept** and **Apply to line**

and click near the following coordinates to select sides to be changed.

2, 0.1
 1, 0.9
 7, 0.1
 7, 6.9
 9, 5
 10, 5
 4, 1
 3, 1

Return to close the submenu.

As the mouse is clicked near each line, the subdivisions will be seen to change.

To select the final two edges, again resize the display:

Redraw Picture → **Numerical limits**

Minimum on horizontal axis	=	0
Maximum on horizontal axis	=	50
Minimum on vertical axis	=	0
Maximum on vertical axis	=	50
<input type="button" value="Accept"/>		<input type="button" value="Dismiss"/>

Choose the **Set Subdivision**

Subdivision		
<input type="checkbox"/> 1	<input type="checkbox"/> 2	<input type="checkbox"/> 3
<input type="checkbox"/> 4	<input type="checkbox"/> 5	<input type="checkbox"/> 6
<input type="checkbox"/> 7	<input checked="" type="checkbox"/> 8	<input type="checkbox"/> 9
Other	<input type="text"/>	
<input type="button" value="Accept"/>		

Select **Accept** and **Apply to line**

and click near the following coordinates to select sides to be changed.

30, 0.1
 30, 29

Return to close the submenu.

Reset the display to its original size using:

... Redraw Picture → Previous size

This completes the definition of the baseplane. It is important that the structure of the baseplane is correct. Once the baseplane has been defined and extrusions created, no facet can be deleted, although point coordinates can be altered. The completed baseplane has been shown already in Figure 9.3.

To proceed with the next step of building a FE-model, close the **Subdivisions** menu with **Return**.

Extruding in the Third Dimension

The model being built will have 10 extrusion layers.

In summary these are

Z=	Description	Layer
Air layers		
–50 to –10	First layer of air outside the speaker.	1
–10 to –5	Next layer of air up to the base of the speaker	2
Speaker section		
–5 to –4	Base plate of the speaker	3
–4 to –1	Cylindrical centre pole and sides	4
–1 to 2	Magnet within the centre pole	5
2 to 4	Centre pole thins	6
4 to 5	Top plate of the speaker, and air gap between centre pole and this plate.	7
Air layers		
5 to 6	Air layer above the top of the speaker.	8
6 to 10	Intermediate air layer	9
10 to 50	Air layer extending out to the far-field boundary	10

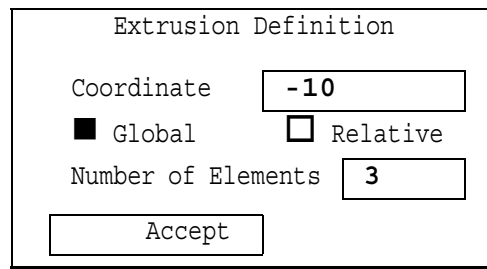
The baseplane has been defined to lie at $Z=-50$, so the first extrusion will be linear to -10 (i.e. the first air layer). Quadratic extrusions are used when it is necessary to create a curved surface in the third dimension, e.g. if creating a sphere, as a mid-layer of nodes is created between the new and the previous plane. They may also be used to grade the mesh in the third direction.

Select **Extrude** from the **Define Baseplane** menu, then select **Linear Extrusion**.

The extrusion can be performed by either a *Global* extrusion to $Z=-10$, or by a *Relative* extrusion of 40. Either of the extrusions (in this example¹) will have the same effect: the global extrusion will create a duplicate of the baseplane with $Z=-10$, the relative extrusion will create a duplicate of the baseplane by moving all points by 40 units in the Z direction.

The number of elements refers to the number of elements that the extruded sides will be divided into when creating the finite element mesh for analysis.

Complete the dialog box as shown



The dialog box titled "Extrusion Definition" contains the following fields and options:

- Coordinate:** A text box containing the value **-10**.
- Global:** A radio button that is selected (indicated by a filled square).
- Relative:** An unselected radio button (indicated by an empty square).
- Number of Elements:** A text box containing the value **3**.
- Accept:** A button at the bottom of the dialog.

and select **Accept**.

A message box appears stating **Layer 1 created**. This is cleared by pressing **Continue**.

You are now be given the option to modify point coordinates of the new plane that has been created by the extrusion, of setting material properties and of setting boundary conditions for baseplane, top plane and extrusion facets. All of these will be modified later, once all extrusions have been completed, so **Finish editing** should be pressed, followed by **Finish** for each subsequent menu (4 times in total) until the top level command menu is reached.

The model may now be saved, should it be necessary to end the session or as a safety backup. Select

FILE ↓

Write pre-processor file

and enter **speaker** in the **Selected file** box and press **Accept**. The file speaker.oppre has now been saved.

1. It is not necessary that all points in the baseplane have the same W coordinates i.e. the points do not need to be coplanar. When this is the case, the relative and global extrusions will have different effect.

Each time the mesh is extended, a new layer of points is created by duplicating the position of points from the previous layer. To continue with the extrusion process¹ select

DEFINE ↓

Extend existing mesh → Extend without editing

Mesh number	=	1
Accept		Dismiss

XY plane, extrude in Z

A box similar to the previous extrusion box can now be seen. In each case *Global* extrusions will be used. For each box select the **Accept** and **extend again** box until the final extrusion is reached Z=50, when the **Accept, this is the last** button should be used.

Extrusion Definition	
Coordinate	-5
<input checked="" type="checkbox"/> Global	<input type="checkbox"/> Relative
<input checked="" type="checkbox"/> Linear	<input type="checkbox"/> Quadratic
Number of Elements	3
Accept - and extend again	
Accept - this is the last	

Extrusion Definition	
Coordinate	-4
<input checked="" type="checkbox"/> Global	<input type="checkbox"/> Relative
<input checked="" type="checkbox"/> Linear	<input type="checkbox"/> Quadratic
Number of Elements	4
Accept - and extend again	
Accept - this is the last	

1. Extrusions create a duplicate of the last mesh plane, so a relative extrusion will duplicate points, changing the W coordinate by the extrusion distance, a global extrusion will duplicate the points, setting the W coordinate of all of them to the new value

Extrusion Definition

Coordinate

☒ Global ☐ Relative

☒ Linear ☐ Quadratic

Number of Elements

Accept - and extend again

Accept - this is the last

Extrusion Definition

Coordinate

☒ Global ☐ Relative

☒ Linear ☐ Quadratic

Number of Elements

Accept - and extend again

Accept - this is the last

Extrusion Definition

Coordinate

☒ Global ☐ Relative

☒ Linear ☐ Quadratic

Number of Elements

Accept - and extend again

Accept - this is the last

Extrusion Definition

Coordinate

☒ Global ☐ Relative

☒ Linear ☐ Quadratic

Number of Elements

Accept - and extend again

Accept - this is the last

Extrusion Definition	
Coordinate	<input type="text" value="6"/>
<input checked="" type="checkbox"/> Global	<input type="checkbox"/> Relative
<input checked="" type="checkbox"/> Linear	<input type="checkbox"/> Quadratic
Number of Elements	<input type="text" value="7"/>
<input type="button" value="Accept - and extend again"/>	
<input type="button" value="Accept - this is the last"/>	

Extrusion Definition	
Coordinate	<input type="text" value="10"/>
<input checked="" type="checkbox"/> Global	<input type="checkbox"/> Relative
<input checked="" type="checkbox"/> Linear	<input type="checkbox"/> Quadratic
Number of Elements	<input type="text" value="4"/>
<input type="button" value="Accept - and extend again"/>	
<input type="button" value="Accept - this is the last"/>	

For the final extrusion select **Accept – this is the last**

Extrusion Definition	
Coordinate	<input type="text" value="50"/>
<input checked="" type="checkbox"/> Global	<input type="checkbox"/> Relative
<input checked="" type="checkbox"/> Linear	<input type="checkbox"/> Quadratic
Number of Elements	<input type="text" value="3"/>
<input type="button" value="Accept - and extend again"/>	
<input checked="" type="button" value="Accept - this is the last"/>	

Select **Return** twice to complete the extrusions of the model.

Making Modifications to the Model

Material Properties

Now the material properties of the model will be specified. The default settings for all the regions is a **TOTAL** potential type with the name **AIR**. The different parts of the model will each be given a different name. Although the material

properties will be the same when generating the analysis file, it assists in viewing the model.

Properties to be set are

Material Name	Potential Type	Layers	Regions
Base	TOTAL	3	All regions with $X \leq 10\text{mm}$
Side	TOTAL	4, 5, 6	The region between $X=9\text{mm}$ and $X=10\text{mm}$
Top	TOTAL	7	The region from 4mm radius up to 10mm
Air	REDUCED	6, 7	Air gap region between 3 and 4mm radius
		8	Both regions within the 4 mm radius
Centre	TOTAL	4, 6, 7	The region within the 3mm radius
Magnet	TOTAL	5	The region within the 3mm radius

For each of the materials listed in the table above, the regions should be set to the properties shown above. To set the material properties for the base, select

MODIFY ↓

Material properties

Layer number	=	3
<input type="button" value="Accept"/>		<input type="button" value="Dismiss"/>

Select/de-select volume

0.5, 0.2
3.5, 1
6.5, 1

Select and define

9.5, 1

All the regions within the 10mm layer should now be highlighted. The menu box appears and should be filled in as shown below.

Material Definition			
Material Name	Base		
Potential Type:	Element Type:		
<input checked="" type="checkbox"/> Total Scalar	<input checked="" type="checkbox"/> Linear		
<input type="checkbox"/> Reduced Scalar	<input type="checkbox"/> Quadratic		
<input type="checkbox"/> Vector			
Options:			
Jx, Jy, Jz			
Vx, Vy, Vz			
Scalar: Charge Density or Rotational Velocity			
Scalar			
Packing factor			
Material orientation			
<input type="checkbox"/> Local XYZ=XYZ	<input type="checkbox"/> Local XYZ=YZX	<input type="checkbox"/> Local XYZ=ZXY	
Other vector			
Other volumes and layers:			
From		To	<input type="checkbox"/> All volumes
Accept	Keep	Help	Quit

followed by **Accept** and **Finish**.

The selected facets turn dark blue, indicating that they are of a certain material type. When defined each different material type will be displayed using a different colour.

The **FROM** and **TO** items in the menu box can be used to set the properties of several layers. e.g.

MODIFY ↓

Material properties

Layer number	=	4
Accept	Dismiss	

Select and define

9.5, 1

Material Definition	
Material Name	<input type="text" value="Side"/>
Potential Type:	Element Type:
<input checked="" type="checkbox"/> Total Scalar	<input checked="" type="checkbox"/> Linear
<input type="checkbox"/> Reduced Scalar	<input type="checkbox"/> Quadratic
<input type="checkbox"/> Vector	
Options:	
Jx, Jy, Jz	<input type="text"/>
Vx, Vy, Vz	<input type="text"/>
Scalar: Charge Density or Rotational Velocity	
Scalar	<input type="text"/>
Packing factor	<input type="text"/>
Material orientation	
<input type="checkbox"/> Local XYZ=XYZ	<input type="checkbox"/> Local XYZ=YZX <input type="checkbox"/> Local XYZ=ZXY
Other vector	<input type="text"/>
Other volumes and layers:	
From <input type="text" value="4"/>	To <input type="text" value="6"/> <input type="checkbox"/> All volumes
<input type="button" value="Accept"/>	<input type="button" value="Keep"/> <input type="button" value="Help"/> <input type="button" value="Quit"/>

Accept and Finish.

The top plate can be set similarly, i.e.

MODIFY ↓

Material properties

Layer number	=	<input type="text" value="7"/>
<input type="button" value="Accept"/>	<input type="button" value="Dismiss"/>	

Select/de-select volume

6.5, 1

Select and define

9.5, 1

Regions from 4mm to the 10mm layer should now be highlighted. The menu box appears and should be filled in as shown below.

Material Definition			
Material Name	Top		
Potential Type:	Element Type:		
<input checked="" type="checkbox"/> Total Scalar	<input checked="" type="checkbox"/> Linear		
<input type="checkbox"/> Reduced Scalar	<input type="checkbox"/> Quadratic		
<input type="checkbox"/> Vector			
Options:			
Jx, Jy, Jz	<input type="text"/>		
Vx, Vy, Vz	<input type="text"/>		
Scalar: Charge Density or Rotational Velocity			
Scalar	<input type="text"/>		
Packing factor	<input type="text"/>		
Material orientation			
<input type="checkbox"/> Local XYZ=XYZ	<input type="checkbox"/> Local XYZ=YZX	<input type="checkbox"/> Local XYZ=ZXY	
Other vector	<input type="text"/>		
Other volumes and layers:			
From	<input type="text"/>	To	<input type="text"/> <input type="checkbox"/> All volumes
<input type="button" value="Accept"/>	<input type="button" value="Keep"/>	<input type="button" value="Help"/>	<input type="button" value="Quit"/>

Accept and Finish.

Note that it is possible to set the material properties in a layer that is not being displayed. The order is necessary as regions will be set and then redefined. This allows the **FROM** and **TO** options to be used to efficiently in the modification process.

It is necessary to change the potential type of some of the air, as conductors *must* be placed within a reduced potential volume.

MODIFY ↓

Material properties

Layer number	=	7
<input type="button" value="Accept"/>	<input type="button" value="Dismiss"/>	

Select and define

3.5, 1

Material Definition			
Material Name	<input type="text" value="Air"/>		
Potential Type:	Element Type:		
<input type="checkbox"/> Total Scalar	<input checked="" type="checkbox"/> Linear		
<input checked="" type="checkbox"/> Reduced Scalar	<input type="checkbox"/> Quadratic		
<input type="checkbox"/> Vector			
Options:			
Jx, Jy, Jz	<input type="text"/>		
Vx, Vy, Vz	<input type="text"/>		
Scalar: Charge Density or Rotational Velocity			
Scalar	<input type="text"/>		
Packing factor	<input type="text"/>		
Material orientation			
<input type="checkbox"/> Local XYZ=XYZ	<input type="checkbox"/> Local XYZ=YZX	<input type="checkbox"/> Local XYZ=ZXY	
Other vector	<input type="text"/>		
Other volumes and layers:			
From	<input type="text"/>	To	<input type="text"/> <input type="checkbox"/> All volumes
<input type="button" value="Accept"/>	<input type="button" value="Keep"/>	<input type="button" value="Help"/>	<input type="button" value="Quit"/>

Accept.

The centre pole will be created

Select and define

2, 0.5

and complete the parameter box:

Material Definition			
Material Name	<input type="text" value="Centre"/>		
Potential Type:	Element Type:		
<input checked="" type="checkbox"/> Total Scalar	<input checked="" type="checkbox"/> Linear		
<input type="checkbox"/> Reduced Scalar	<input type="checkbox"/> Quadratic		
<input type="checkbox"/> Vector			
Options:			
Jx, Jy, Jz	<input type="text"/>		
Vx, Vy, Vz	<input type="text"/>		
Scalar: Charge Density or Rotational Velocity			
Scalar	<input type="text"/>		
Packing factor	<input type="text"/>		
Material orientation			
<input type="checkbox"/> Local XYZ=XYZ	<input type="checkbox"/> Local XYZ=YZX	<input type="checkbox"/> Local XYZ=ZXY	
Other vector	<input type="text"/>		
Other volumes and layers:			
From	<input type="text" value="4"/>	To	<input type="text" value="7"/> <input type="checkbox"/> All volumes
<input type="button" value="Accept"/>	<input type="button" value="Keep"/>	<input type="button" value="Help"/>	<input type="button" value="Quit"/>

Keep

which sets the material type, and keeps the presently selected facets selected. After having cleared the message box, the next assignment can be made immediately by completing the parameter box:

Material Definition			
Material Name	Magnet		
Potential Type:	Element Type:		
<input checked="" type="checkbox"/> Total Scalar	<input checked="" type="checkbox"/> Linear		
<input type="checkbox"/> Reduced Scalar	<input type="checkbox"/> Quadratic		
<input type="checkbox"/> Vector			
Options:			
Jx, Jy, Jz	<input type="text"/>		
Vx, Vy, Vz	<input type="text"/>		
Scalar: Charge Density or Rotational Velocity			
Scalar	<input type="text"/>		
Packing factor	<input type="text"/>		
Material orientation			
<input type="checkbox"/> Local XYZ=XYZ	<input type="checkbox"/> Local XYZ=YZX	<input type="checkbox"/> Local XYZ=ZXY	
Other vector	<input type="text"/>		
Other volumes and layers:			
From	5	To	5 <input type="checkbox"/> All volumes
<input type="button" value="Accept"/>	<input type="button" value="Keep"/>	<input type="button" value="Help"/>	<input type="button" value="Quit"/>

Accept, Finish and Return.

The default orientation of the magnetisation vector in the material MAGNET is the z-direction. In the present model this is the required direction, so no changes are needed.

To see the wire frame model, select:

DISPLAY ↓

3d viewer Refresh display

This activates the OpenGL window allowing the user to rotate, zoom and translate the view. The left mouse button controls the view, while the right mouse button allows the view function to be changed.

A view of the model, as shown in Figure 9.4. should be obtained. **Return** to close the **Display** menu.

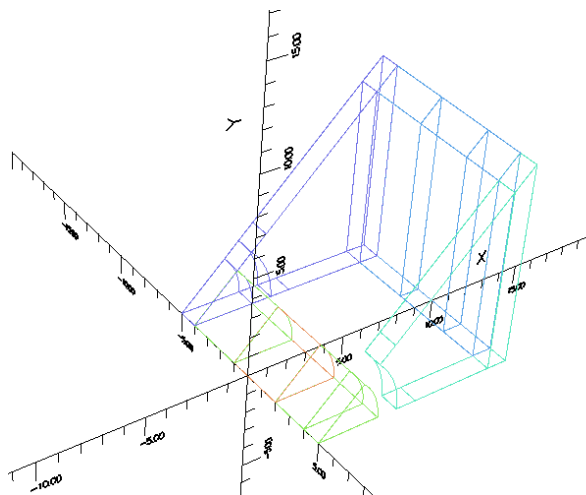


Figure 9.4 The model so far

Modifying Point Coordinates

It is required that the top of the CENTRE be tapered to a radius of 2mm, and the top plate has an inner radius of 2.5mm. To achieve this shape it is necessary to modify the point coordinates of mesh planes 7 and 8 (i.e. the planes below and above mesh layer 7, containing the top plate).

To do this points will be moved such that the outer points on the centre pole are scaled towards (0,0) by a factor of $2/3$ so that they move from their current position on a radius of 3mm to a new position at a radius of 2mm. The points on the inner radius of the top plate will be scaled by a factor of $2.5/4$, to move them from a radius of 4mm to 2.5mm.

Select

MODIFY ↓

Point Coordinates

Plane number	=	7
<input type="button" value="Accept"/>		<input type="button" value="Dismiss"/>

Select/de-select point

3, 0

2.8, 1

2.1, 2.1

Transform points → Scale

U Coordinate	=	0
V Coordinate	=	0
Factor	=	2/3
<input type="button" value="Accept"/>		<input type="button" value="Dismiss"/>

Return

Select/de-select point

4, 0
 3.6, 1.5
 2.9, 2.9

Transform points → Scale

U Coordinate	=	0
V Coordinate	=	0
Factor	=	2.5/4
<input type="button" value="Accept"/>		<input type="button" value="Dismiss"/>

Return

Finish editing

The same process must now be repeated for mesh plane 8, where points (3,0), (2.8,1), (2.1,2.1) are scaled by a factor 2/3, and points (4,0), (3.6,1.5) and (2.9,2.9) are scaled by a factor 2.5/4.

This completes the point transformations that are necessary for this model. Figure 9.5 shows the modified design.

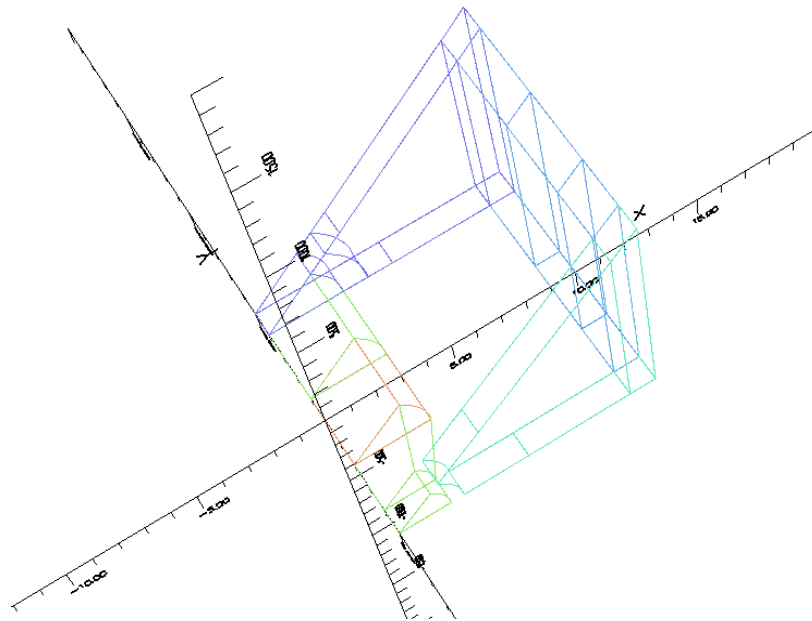


Figure 9.5 The model complete with modifications

Boundary Conditions

The boundary conditions that will be set are **TANGENTIAL MAGNETIC** on the three sides of the 45° wedge, and the top and bottom of the model i.e. at $Z=\pm 50$. Since no **NORMAL MAGNETIC** boundary condition is assigned, it is left to TOSCA to gauge the model, by arbitrarily setting the potential to zero at one node in the model.

To set all external facets simultaneously is a simple procedure, as follows. Select:

MODIFY ↓

Boundary conditions → Base plane → All external facets

and complete the following parameter box:

Global Boundary Condition Definition		
Condition name:		
<input type="checkbox"/> Magnetic Scalar	<input type="checkbox"/> Normal Magnetic	<input checked="" type="checkbox"/> Tangential Magnetic
<input type="checkbox"/> Voltage	<input type="checkbox"/> Normal Electric	<input type="checkbox"/> Tangential Electric
<input type="checkbox"/> Total Ax	<input type="checkbox"/> Total Ay	<input type="checkbox"/> Total Az
<input type="checkbox"/> Incident Ax	<input type="checkbox"/> Incident Ay	<input type="checkbox"/> Incident Az
<input type="checkbox"/> Incident Voltage	<input type="checkbox"/> Perfect Conductor	<input type="checkbox"/> Radiation
<input type="checkbox"/> Normal Derivative	<input type="checkbox"/> Mixed Derivative	
<input type="checkbox"/> Symmetry	<input type="checkbox"/> Slip Surface	<input type="checkbox"/> Clear
Value	<input type="text"/>	Label/2nd value <input type="text"/>
<input type="button" value="Accept"/>	<input type="button" value="Help"/>	<input type="button" value="Quit"/>

Accept.

This completes the finite element mesh structure needed for the model, so select **Finish**, followed by **Return** twice.

Modify the Subdivision

At present, each plane through the model has the same subdivision. This is wasteful, since the base plane and the top plane, both removed from the region of interest, do not require such a refined mesh.

It is possible to reduce the discretisation on these two planes.

Select:

MODIFY ↓

Subdivisions → Variable subdivision → In plane

and select plane number 1. Initially, set all lines on the base plane to have a subdivision of 2, by selecting **Select and define** and choose any edge. Then complete the following parameter box:

Subdivision Definition			
No. of elements	<input type="text" value="2"/>		
Other planes or layers:		All lines:	
From	<input type="text" value="1"/>	To	<input type="text" value="2"/> <input checked="" type="checkbox"/> All
<input type="button" value="Accept"/>		<input type="button" value="Keep"/> <input type="button" value="Help"/> <input type="button" value="Quit"/>	

Set some edges to a subdivision of 4 by **Select/de-select line** and selecting the following lines:

2.8, 1
 3.8, 1
 6, 0.1
 6, 6
 9, 5
 10, 5
 30, 0.1
 30, 29

and then **Select and define** and select the line:

49, 25

and complete the parameter box:

Subdivision Definition			
No. of elements	<input type="text" value="4"/>		
Other planes or layers:		All lines:	
From	<input type="text" value="1"/>	To	<input type="text" value="2"/> <input type="checkbox"/> All
<input type="button" value="Accept"/>		<input type="button" value="Keep"/> <input type="button" value="Help"/> <input type="button" value="Quit"/>	

Repeat the procedure for layer 10, which is the top air layer. Layer 10 is situated between plane 10 and plane 11.

So select **Finish** once, and then **In plane** and select plane 10, and repeat the above subdivision definitions in plane 10, using the **FROM** and **TO** option (from 10 to 11). Complete this by selecting **Finish**, and then **Return** three times.

Conductor Definition

The next stage involves adding a conductor that will sit within the air gap. This will be a solenoid of thickness 2 mm and height 6 mm placed centrally within the air gap. The parameters of the conductor definitions are described in the Reference Manual. Select

Define ↓

Conductors → Define a
conductor → Generally
orientated set → Solenoid

Local coord 1: X - origin	=	0
Local coord 1: Y - origin	=	0
Local coord 1: Z - origin	=	0
<input type="button" value="Accept"/>		

Local Coordinate system 1	
XYZ local = XYZ global	<input checked="" type="checkbox"/>
XYZ local = YZX global	<input type="checkbox"/>
XYZ local = ZXY global	<input type="checkbox"/>
Other system	<input type="checkbox"/>
Return	←

Local coord 2: X - origin	=	0
Local coord 2: Y - origin	=	0
Local coord 2: Z - origin	=	4.5
<input type="button" value="Accept"/>		

Local Coordinate system 2	
XYZ local = XYZ global	<input type="checkbox"/>
XYZ local = YZX global	<input checked="" type="checkbox"/>
XYZ local = ZXY global	<input type="checkbox"/>
Other system	<input type="checkbox"/>
Return	←

Cross-section: X1	=	2.15
Cross-section: Y1	=	-0.3
Cross-section: X2	=	2.35
Cross-section: Y2	=	-0.3
<input type="button" value="Accept"/>		

Cross-section: X3	=	2.35
Cross-section: Y3	=	0.3
Cross-section: X4	=	2.15
Cross-section: Y4	=	0.3
<input type="button" value="Accept"/>		

Curvature: CU1	=	0
Curvature: CU2	=	0
Curvature: CU3	=	0
Curvature: CU4	=	0
<input type="button" value="Accept"/>		

Current density	=	50
Symmetry code	=	1
Drive label	=	ONE
<input type="button" value="Accept"/>		

There will be no reflections of this conductor activated, so press **Return**.

Tolerance on flux density	=	0.01
<input type="button" value="Accept"/>		

Close all menus by pressing **Return** four times. This completes the pre processing required to build the 3d model of the speaker.

Before displaying the model and building the database, the surface and volume mesh must be created. These are carried as before, selecting:

MESH ↓
...triangles

followed by

MESH ↓
Mesh

To display the surface discretisation of the model, select

DISPLAY ↓
 3D viewer... ...style

and select the following settings of the subsequent dialog box:

3d Viewer Style	
No elements	<input type="checkbox"/>
Surface elements	<input checked="" type="checkbox"/>
Volume elements	<input type="checkbox"/>
Vectors...	<input type="checkbox"/>
... no vectors	<input type="checkbox"/>
... in conductors only	<input checked="" type="checkbox"/>
... material orientation	<input type="checkbox"/>
... current density	<input type="checkbox"/>
... velocity	<input type="checkbox"/>
Refresh display	*
Return	←

Finish by selecting **Refresh display**, showing the conductor placed within the reduced potential air gap as illustrated in Figure 9.6

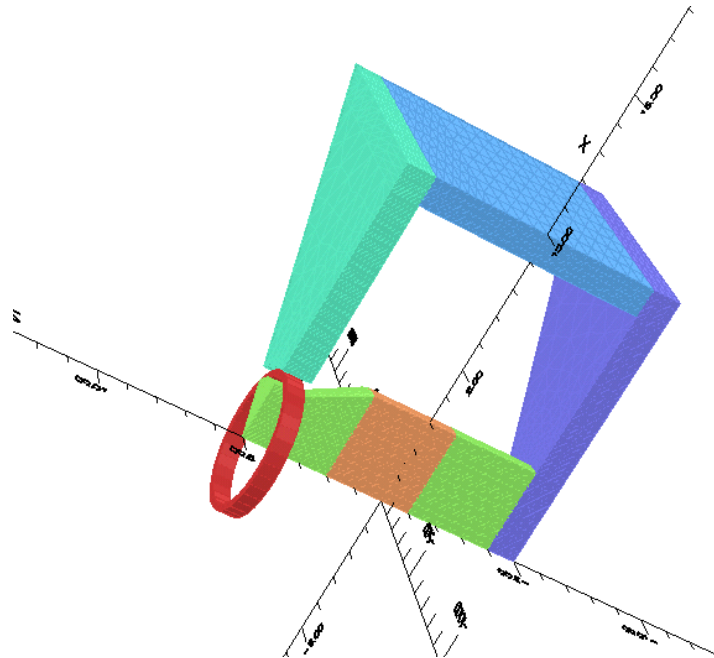


Figure 9.6 The model including the solenoid

Creating the TOSCA Analysis Database File

Using the TOSCA analysis module to solve the solenoid model as a static solution, select

FILE ↓

Analysis

... create new database → Statics (TOSCA)

Magnetostatics

and select the units and element type to be used by completing the parameter box as shown below:

New Analysis Database File

File ↓

Units: ☒ CGS ☐ SI (metres) ☐ SI (mm) ☐ SI (microns) ☐ Inches

Element type: ☒ Linear ☐ Quadratic ☐ Mixed

followed by **Accept**.

If the box ↓ alongside the file name is selected, a file selection box is raised, allowing all existing files to be shown/selected.

After a short pause whilst the main database is prepared, a further parameter box is presented, allowing a title to be specified for the model. Any number of lines of text can be input, and the title is terminated by typing a single * character on a line on its own. A message then appears indicating that the simulation has been successfully created.

Clear the message box by clicking **Continue**.

The TOSCA (Magnetic) data menu now appears

TOSCA (Magnetic)	
Materials	→
Linear solution	■
Non-linear solution	<input type="checkbox"/>
Adaptive RHS Integrals	<input type="checkbox"/>
Periodicity conditions	→
External fields	→
Add drive fields	→
Automatic potential cuts	<input type="checkbox"/>
Check data	*
Return	←

Initially, select **Materials**.

For each material in the model, the linear or non-linear characteristics can be defined. In the present model, the following parameter box will appear. Set the non-linear button, and then select material **BASE** and **Define**.

Material Names	
BASE	Properties of air
SIDE	Properties of air
TOP	Properties of air
CENTRE	Properties of air
MAGNET	Properties of air

Material Characteristics

☐ Linear
 ☒ Non-linear

☒ Isotropic
 ☐ Packed
 ☐ Multiple

Since this is a non-linear material, the properties for material BASE must be given.

Properties for material BASE

Non-linear BH curve

Isotropic

The program expects the non-linear material data (BH data.) to be given in a file with the extension *.bh*. This file can be selected using the ↓ box. Alternatively, there is a button, which provides a “built in” BH curve, also called Default BH curve.

The materials BASE, SIDE, TOP and CENTRE will each use the “*default.bh*” curve, so select **Default BH**, and repeat the same exercise for the other materials.

Material type MAGNET uses a different BH curve, as this is a permanent magnet, and the BH curve will include a coercive force value. The BH file to use is “*alcomax3.bh*”, and it is in a sub-directory included with the software.

Selecting the ↓ box brings up a file selection box:

File Selection Box

Filter

Files in: current path Subdirectories

Selected file

Selected directory

Now the BH file “*alcomax3.bh*” is to be selected. At first the directory has to be changed. In the “selected directory” box, type one of the following pathnames.

UNIX operating system, use:

\$vfdir/bh

and on Windows use:

%vfdir%\bh

Hitting return (or **CD**) will change to that directory, and will show a selection of BH curves to choose from.

Select "*alcomax3.bh*" and **Accept** the selection. This will close the file selection box, and automatically insert the filename into the material properties box. **Accept** this selection.

Now all materials have been defined. Close the **Material names** box with **Return**.

Then select **Non-linear solution**, which immediately brings up the parameter box as follows:

TOSCA nonlinear iteration data	
<input checked="" type="checkbox"/> Newton-Raphson	
<input type="checkbox"/> Simple Update	
Number of iterations	<input type="text" value="21"/>
Convergence tolerance	<input type="text" value="0.001"/>
Underrelaxation factor	<input type="text" value="1"/>
<i>NL iteration per timestep</i>	<input type="text"/>
<input type="button" value="Accept"/>	<input type="button" value="Dismiss"/>

and **Accept**.

Select **Return** in the **TOSCA (Magnetic)** data menu, and check that the data is correct in the information box (note that the exact number of nodes and tetrahedra may vary from that shown below). Close the information box, to complete storing the analysis database.

Opening database speaker.op3, simulation number 1 on 30/Oct/2001
at 17:07:28

File: speaker.op3 simulation: 1
Created on: 30/Oct/2001 17:01:17
In Directory:
My Work Folder
By Machine:
Node: MyComputer. Processor: x86. System: Windows
Log Files: Opera3d_Pre_1.backup/log/lp.
Opera-3d/Pre-processor Version 8.5

TOSCA Magnetostatic analysis

User title:
OPERA-3d User Guide Chapter 9 Example

CGS units

1 conductor (current densities in ACM2):
1 Solenoid
Current Densities: 50.0
Adaptive RHS integrals

Boundary Conditions: TANGMAGN

Non linear iteration data
Newton-Raphson Iterations
Maximum Iterations: 21
Tolerance: 1.0E-03
Under-relaxation: 1.0

5 materials defined in the simulation

BASE	: Non-linear isotropic permeability
	: \$default
SIDE	: Non-linear isotropic permeability
	: \$default
TOP	: Non-linear isotropic permeability
	: \$default
CENTRE	: Non-linear isotropic permeability
	: \$default
MAGNET	: Non-linear isotropic permeability
	: ...\\bh\\alcomax3.bh
	: Permanent magnet material

10324 nodes in the model
Only linear elements exist within the model
50861 linear tetrahedra

i

Continue

Writing the Pre Processor Data File

To save the pre processor commands, select

```
FILE ↓
    Write pre-processor file
```

This brings up a File Selection Box.

The filename *speaker.oppre* was saved earlier, so is seen in the file list. This can be overwritten by double-clicking the filename, and accepting the option to overwrite the existing file. Finish by selecting **Continue**.

Leaving the Pre Processor

The data has now been stored correctly. Exit the pre processor so that the analysis module may be used. Do this by selecting

```
FILE ↓
    End OPERA-3d/Pre
```

and confirm with **YES**

The speaker model must then be submitted to TOSCA for analysis.

Note that although it is possible to launch the analysis directly from the pre processor, this does require extra system resources. It may be necessary to close the pre processor and run the analysis separately (see following section).

Running the TOSCA Simulation

UNIX Operating Systems

From within the OPERA-3d environment select the TOSCA option

Option:

tosca

and from the following prompt:

Please give TOSCA database filename (without the .OP3 suffix)

speaker

and choose that the analysis is carried out immediately:

Do you want to run the analysis now or later? (n or l)

n

The analysis then proceeds automatically.

Windows Systems

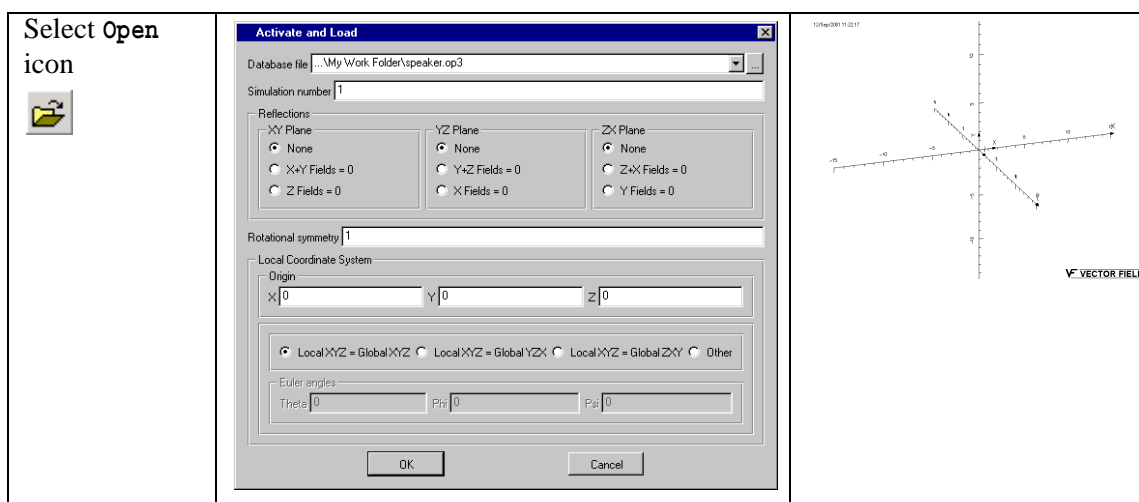
Choose interactive solution under the OPERA-3d menu in the OPERA Console. Select the TOSCA solver and browse to where the *speaker.op3* file was saved and select it as well. Select **OK** to make the analysis module proceed with the calculations.

Post Processing

Start the OPERA-3d post processor.

It is advisable to examine the results from the analysis module and ensure that they correspond with expected results. That is ensure that there are no unusual or non-physical phenomena showing in the results that might indicate errors in the pre processor model. The results of interest are the field through the air gap and the force on the conductor.


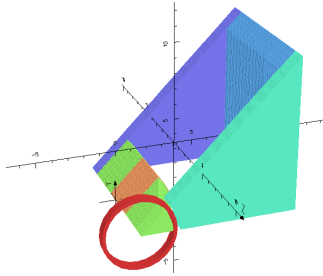
First activate and load the results file. Select




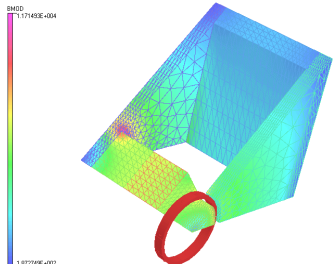

and enter the filename *speaker.op3* or use the ↓ to browse directories and locate the file.

Viewing the Model

Once the file has been activated a display of the flux density on the surface of the model will be examined.

<div>SelectDefault select and refresh icon</div> <div></div>	<div>This will select all materials not called air for display by default. The timer bar shows how the selection and display process is proceeding.</div>	<div></div>
-------------------------------------------------------------------------------------------------------------------------------------------------------	------------------------------------------------------------------------------------------------------------------------------------------------------------------	------------------------------------------------------------------------------------------------

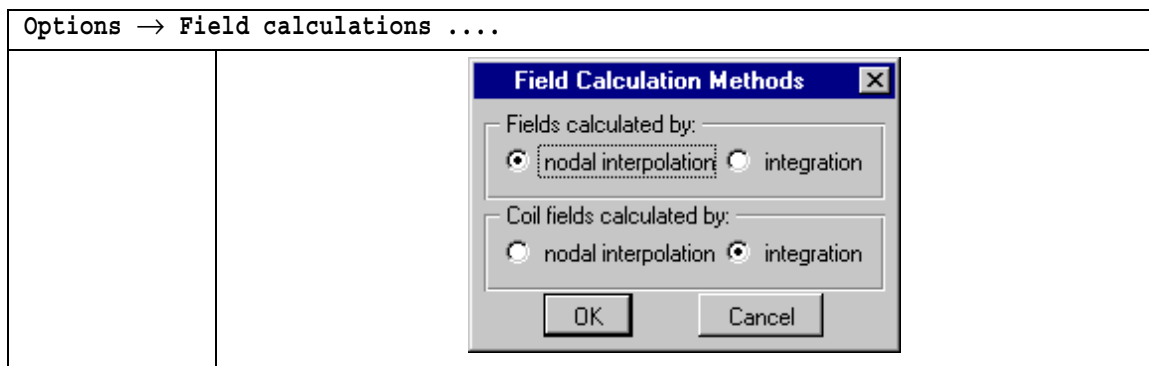
To view the magnitude of the flux density (**BMOD**) over the surface of the model

<div>Select 3D display icon</div> <div></div>	<div><div><div>3d Display</div><div><div><div><div>Material colours</div><div>Component contours</div></div><div>Field component <div>bmod</div></div><div><div>Component limits</div><div><div>Minimum</div><div>Maximum</div></div></div><div><div>Field vectors</div><div>Vector components:<div><div>X</div><div>Y</div><div>Z</div></div></div><div><div>View parameters</div><div><div>Centre of picture</div><div><div>X</div><div>Y</div><div>Z</div></div><div><div>Rotation around axes</div><div><div>X</div><div>Y</div><div>Z</div></div><div><div>Size</div><div>10</div></div><div><div>Perspective view</div></div></div><div><div>OK</div><div>Cancel</div></div></div></div></div><div><div>Change to Component contours and specify the component bmod (it is not necessary to set the View parameters, the defaults can be used).</div></div></div></div></div></div>	<div><div></div></div>
<div>Toggle Axes icon</div> <div></div>	<div>The axes of the coordinate system can be toggled on and off.</div>	

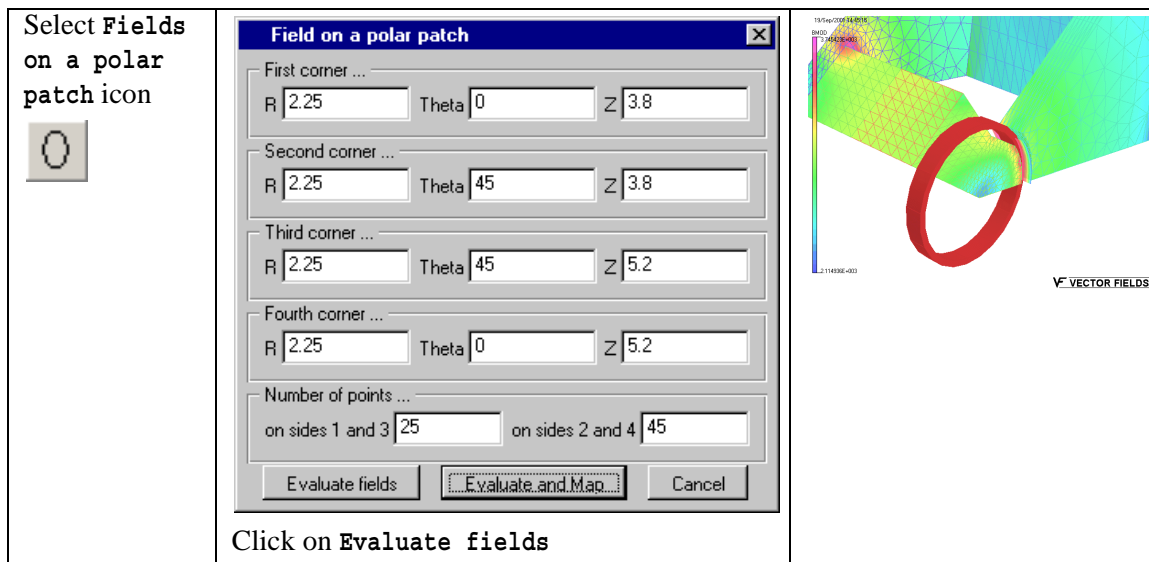
Flux in the Air Gap

To look at the flux in the air gap, a 2D patch or surface (in cylindrical coordinates) will be defined.

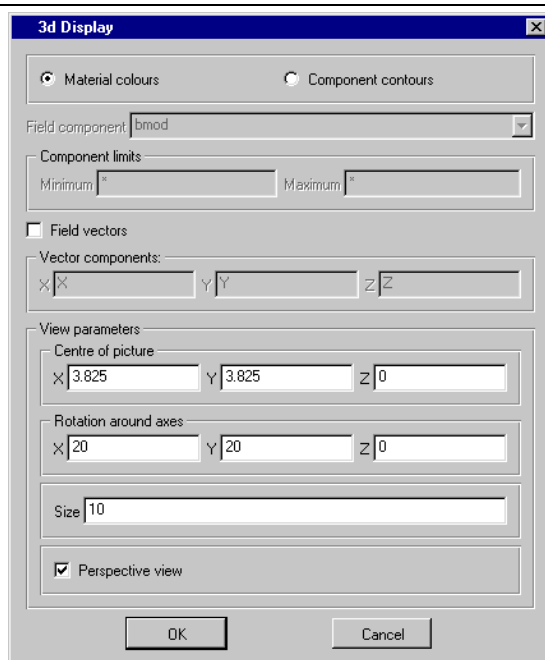
First the calculation method will be changed to improve the field calculation in the reduced potential region. Integral coil field and nodally interpolated fields will be used.



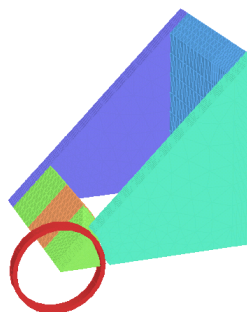
To define the 2D patch in polar coordinates



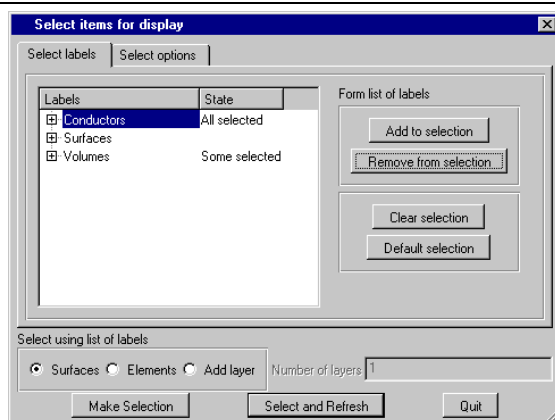
Select 3D display icon



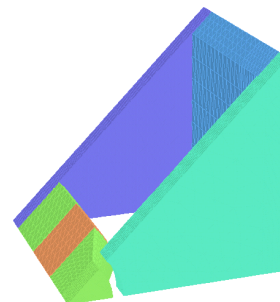
Change to **Material colours**. There is no need to set the **View parameters**.

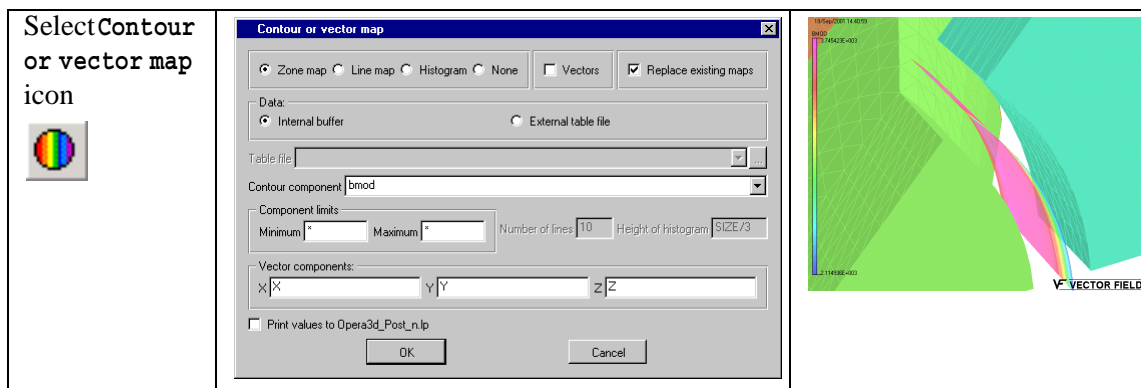


Select the **Select...** icon



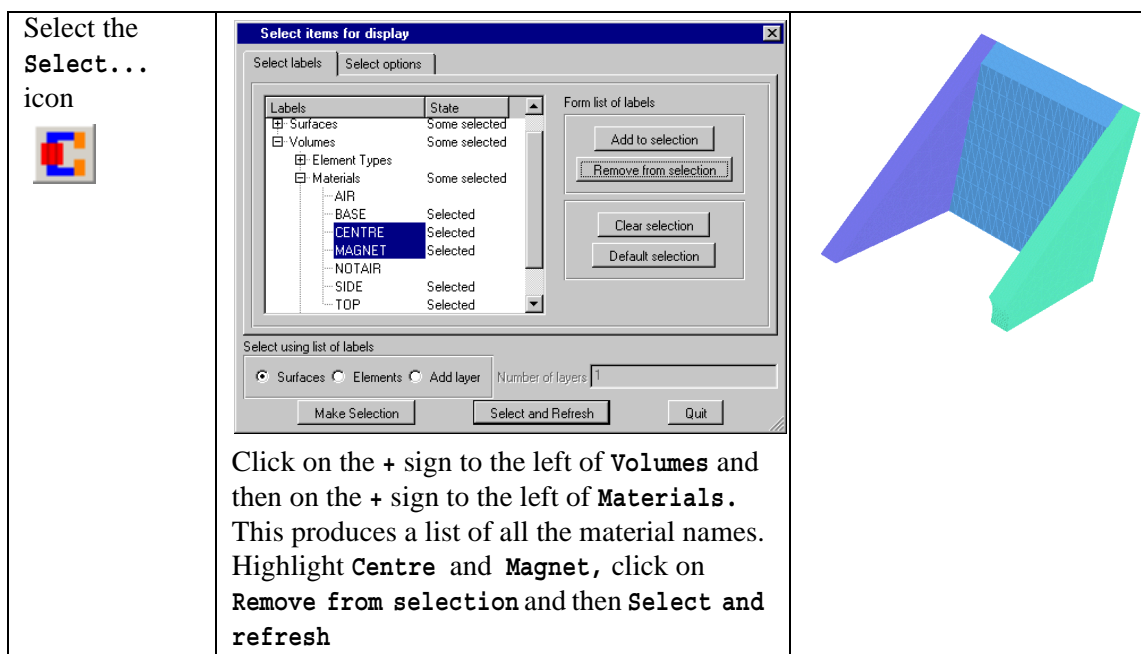
Highlight **Conductor**, click on **Remove from selection** and then **Select and refresh**

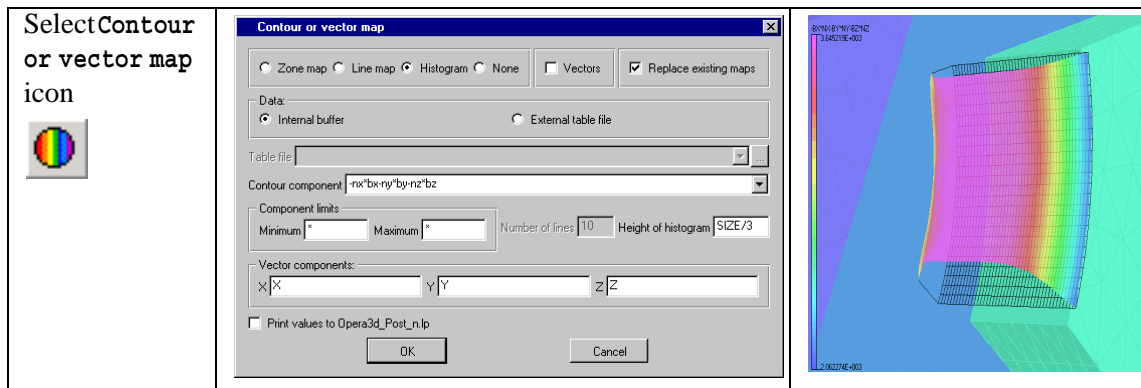




A coloured contour map is drawn in the air gap between the centre pole and the top plate, and it can be seen to be correctly positioned within the air gap. This patch may be hard to see as it is drawn over the geometry already displaying contours. This is improved by redisplaying the model without the contours, removing the coil from the display and zooming in on this air gap.

A histogram (or contoured surface) can be used to view the results more effectively. In addition, part of the geometry will be removed first to enable the histogram to be displayed more clearly.



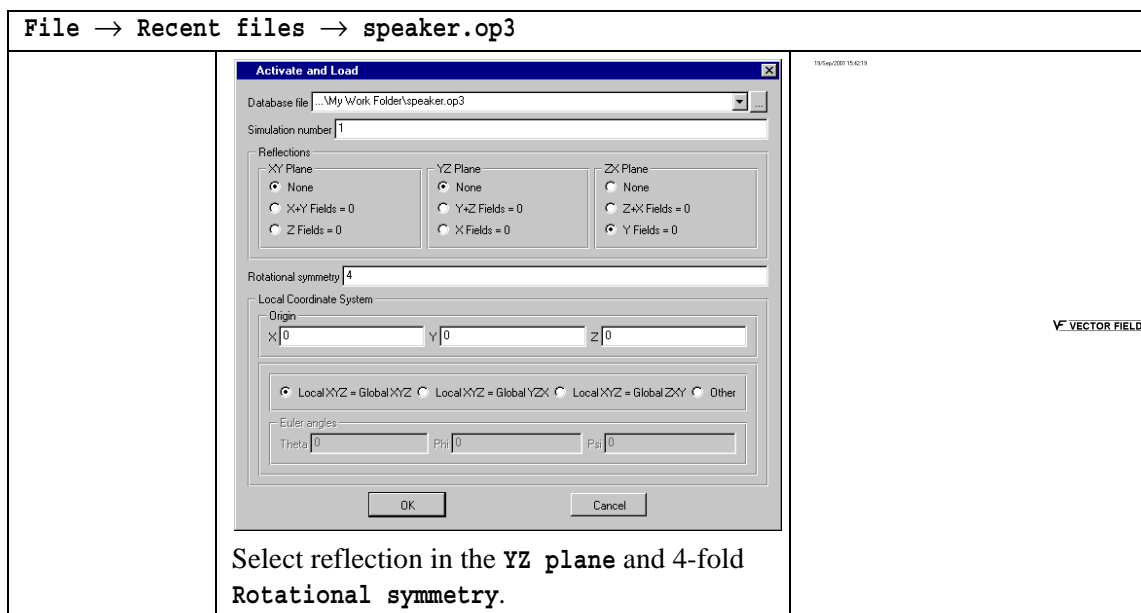


The component that is displayed is the radial flux through the air gap. This is defined, in this particular case, as the negative component of the outward normal to the surface ($-n_x \cdot b_x - n_y \cdot b_y - n_z \cdot b_z$).



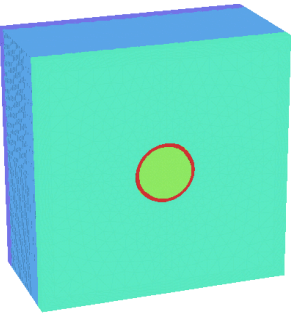
Force on the Conductor

The force on the conductor will be calculated using the Lorentz force equation (i.e. $\mathbf{J} \times \mathbf{B}$).





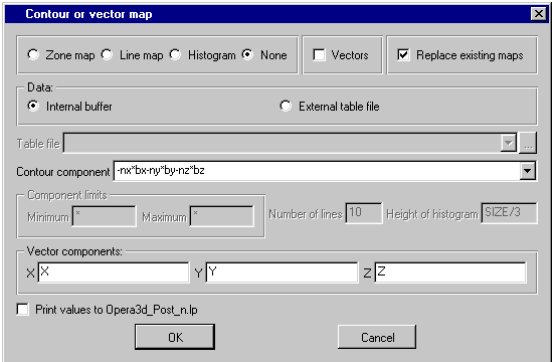
As the conductor lies outside parts of the finite element mesh, it is necessary to reactivate the file but with reflections and rotations so that the full model is present.




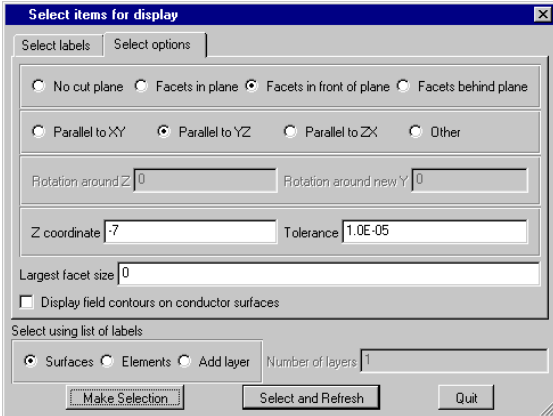

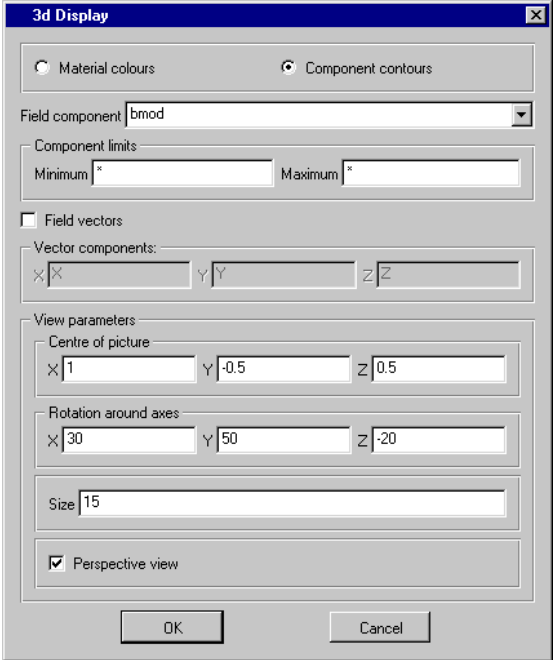
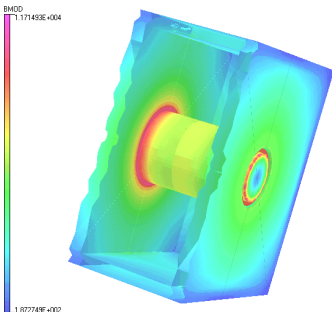

Select reflection in the YZ plane and 4-fold Rotational symmetry.

<p>SelectDefault select and refresh icon</p>  <p>SelectInitial view icon</p> 	<p>The initial view icon allows the complete geometry to be seen.</p>	
-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-----------------------------------------------------------------------	-------------------------------------------------------------------------------------


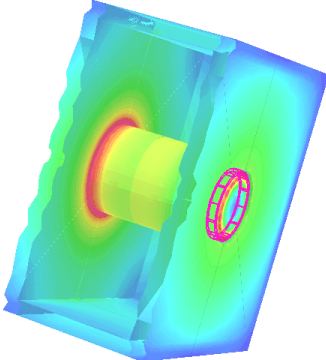
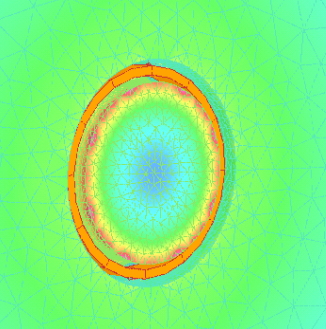



The histogram contour can be hidden or erased:

<p>To temporarily hide the contours: toggle the Contour map icon:</p> 	<p>There are 2 icons with the same design. This one is in the View → Parts of the display toolbar:</p> 	
<p>To erase the contours: select the Contour or vector map icon</p> 	<p>This one is in the Fields toolbar:</p>   <p>Select None.</p>	

The field solution on the complete model can be examined. To look inside the model, a cut plane can be introduced.

<p>Select the Select... icon</p> 	<p>Click on the Select options tab to bring up the dialog box for the cut plane.</p>  <p>Click on Make selection</p>	
<p>Select 3D display icon</p> 		
<p>Toggle the Outline view of model icon</p> 	<p>Turns off the display of the surface mesh.</p>	

To calculate the force on the conductor

<p>Toggle Conductor picking icon</p> 	<p>Move the mouse over the geometry until the coil is highlighted by a pink wire frame outline.</p>	
	<p>Double click to select the coil. The coil turns orange to show it has been selected.</p>	
<p>Select Lorentz forces in conductors icon</p> 		

The fields over the conductor are calculated and the net force found from $\mathbf{J} \times \mathbf{B}$. As would be expected, only the Z-directed force component is significant, with a value of -0.3077.

Summary

This completes this example, and has shown some of the features available in OPERA-3d.

To exit the post processor

```
File↓  
    Exit
```


Chapter 10

3D Levitation Example Using ELEKTRA

Introduction

In this example, consider a magnetic levitation system consisting of an electromagnet suspended above an induction plate. The levitation force is created by the interaction of an a.c. source current flowing in the coil, with a frequency of 60 Hz, and the resulting eddy currents which are induced in the copper plate. A picture of the model is shown in Figure 10.1.

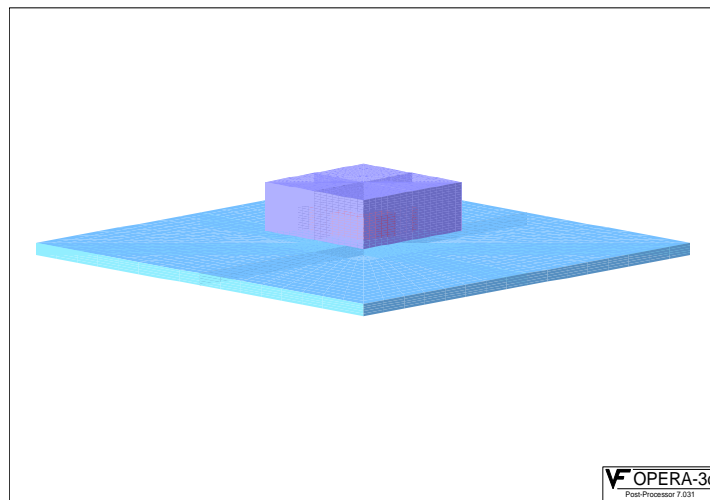


Figure 10.1 The levitating-magnet system to be analyzed. The magnet (top) contains a racetrack coil.

Assuming a given levitation height, the levitation force on the electromagnet and subsequently the source current needed to make the time-averaged net force on the electromagnet equal to its weight will be calculated.

Pre Processing, Constructing the Model

The Baseplane

First draw a 2-dimensional section, then extrude in a third direction to create the full 3-dimensional structure. By exploiting symmetries present in a model, it is possible to minimise the time and computational resources to analyse a model. This particular geometry has three-fold symmetry. Note that the symmetry operations may be restored later during post processing.

Defining the Baseplane Points

Extrusions will occur in the -z direction and the baseplane will serve as the far-field boundary above the magnet:

DEFINE ↓

Define new mesh → Finite element mesh → XY plane,
extrude in Z

and set:

W coordinate of plane = 30

Specify the initial size of the display:

Minimum on horizontal axis	=	0
Maximum on horizontal axis	=	20
Minimum on vertical axis	=	0
Maximum on vertical axis	=	20
<input type="button" value="Accept"/> <input type="button" value="Dismiss"/>		

Accept.

Select Point Input.

Use construction lines to help define the points in the baseplane:

Construction lines → Enter C_Lines → By parameters → Line

Then fill in the parameter boxes as shown, selecting **Accept** after each:

Start.	U	=	0
.....	V	=	0
Finish	U	=	20
.....	V	=	0
Rotation		=	0
<input type="button" value="Accept"/>		<input type="button" value="Dismiss"/>	

Start.	U	=	0
.....	V	=	0
Finish	U	=	20
.....	V	=	20
Rotation		=	0
<input type="button" value="Accept"/>		<input type="button" value="Dismiss"/>	

Start.	U	=	1
.....	V	=	0
Finish	U	=	1
.....	V	=	1
Rotation		=	0
<input type="button" value="Accept"/>		<input type="button" value="Dismiss"/>	

Start.	U	=	2
.....	V	=	0
Finish	U	=	2
.....	V	=	2
Rotation		=	0
<input type="button" value="Accept"/>		<input type="button" value="Dismiss"/>	

Start.	U	=	3
.....	V	=	0
Finish	U	=	3
.....	V	=	3
Rotation		=	0
<input type="button" value="Accept"/>		<input type="button" value="Dismiss"/>	

Start.	U	=	10
.....	V	=	0
Finish	U	=	10
.....	V	=	10
Rotation		=	0
<input type="button" value="Accept"/>		<input type="button" value="Dismiss"/>	

Start.	U	=	20
.....	V	=	0
Finish	U	=	20
.....	V	=	20
Rotation		=	0
<input type="button" value="Accept"/>		<input type="button" value="Dismiss"/>	

Close the construction line menus by pressing **Return** 3 times.

Enter points using: **At C_line intersection**. Points should be defined at the following coordinates:

(0, 0), (1, 0), (2, 0), (3, 0), (10, 0), (20, 0), (1, 1), (2, 2), (3, 3), (10, 10), (20, 20).

Then press **Return**.

Defining the Baseplane Facets

Select **Facet Input** and draw the facets as shown below in Figure 8.2.

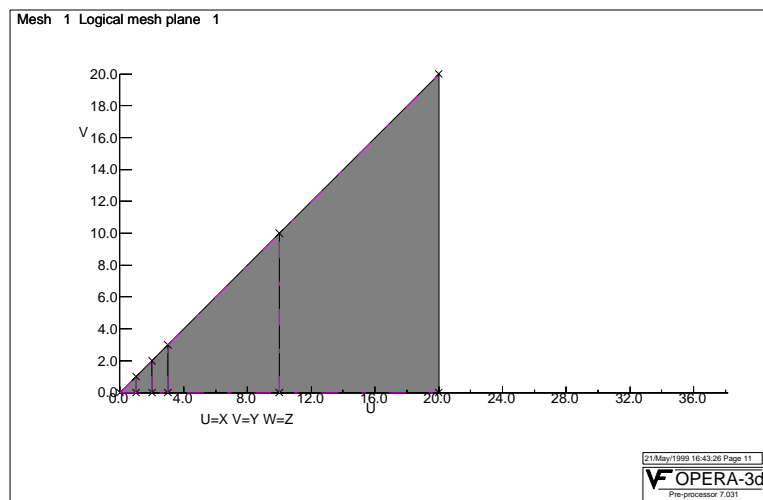


Figure 10.2 Display of the baseplane facets.

There are 4 four-sided facets and 1 three-sided facet - with a vertex located at (0, 0). After having defined the 5 facets, close the **Facet Definition** menu.

Defining the Baseplane Subdivisions

Select **Subdivision** and set the subdivision by selecting **Set subdivision** and complete the parameter box:

Subdivision		
<input type="checkbox"/> 1	<input type="checkbox"/> 2	<input type="checkbox"/> 3
<input type="checkbox"/> 4	<input type="checkbox"/> 5	<input type="checkbox"/> 6
<input type="checkbox"/> 7	<input checked="" type="checkbox"/> 8	<input type="checkbox"/> 9
Other	<input type="text"/>	
<input type="button" value="Accept"/>		

and **Accept**. Next **Apply globally** to set 8 subdivisions along each facet edge. Close the menu with **Return**.

Refine the subdivision distribution using **Set subdivision**

Subdivision		
<input type="checkbox"/> 1	<input type="checkbox"/> 2	<input type="checkbox"/> 3
<input type="checkbox"/> 4	<input type="checkbox"/> 5	<input type="checkbox"/> 6
<input type="checkbox"/> 7	<input type="checkbox"/> 8	<input type="checkbox"/> 9
Other	<input type="text" value="10"/>	
<input type="button" value="Accept"/>		

and **Accept**. Select **Apply to line** and apply the subdivision by clicking near (20, 10).

Press **Return**. Similarly, **Set subdivision**

Subdivision		
<input type="checkbox"/> 1	<input type="checkbox"/> 2	<input type="checkbox"/> 3
<input type="checkbox"/> 4	<input type="checkbox"/> 5	<input type="checkbox"/> 6
<input type="checkbox"/> 7	<input type="checkbox"/> 8	<input type="checkbox"/> 9
Other	<input type="text" value="15"/>	
<input type="button" value="Accept"/>		

Accept followed by **Apply to line** and clicking just above (6, 0). Press **Return**. Finally, **Set subdivision**

Subdivision

<input type="checkbox"/> 1	<input type="checkbox"/> 2	<input checked="" type="checkbox"/> 3
<input type="checkbox"/> 4	<input type="checkbox"/> 5	<input type="checkbox"/> 6
<input type="checkbox"/> 7	<input type="checkbox"/> 8	<input type="checkbox"/> 9

Other

followed by **Accept**, then **Apply to line** and click just above (15, 0). Note that the edges opposite those chosen in the last three steps should also have adjusted. This is necessary so that hexahedral elements may be used.

Close the subdivisions menu by selecting the option **Check for Hex meshing** that confirms the model suitability for hexahedral element generation.

Creating the 3D Model

Defining the Extrusions

Now that the baseplane is complete, the three-dimensional geometry is created by extruding in the -z direction. The table below summarizes what is in each of the 8 layers of this model.

Layer	Material and type of Potential
1,2	air (REDUCED) magnet side
3	magnet (TOTAL) and air (REDUCED)
4	magnet (TOTAL), air (REDUCED), and coil
5	air (REDUCED) between plate and magnet,
6	Plate (VECTOR), air (REDUCED)
7,8	Plate side air (REDUCED)

To create the first layer, select **Extrude** and then select **Linear extrusion** and complete the dialog box as shown

Extrusion Definition

Coordinate

☒ Global ☐ Relative

Number of Elements

and **Accept**.

After clear the message box, select **Finish Editing** followed by **Finish** five times to close all the submenus.

The first extrusion is complete. The subsequent extrusions are created in the following manner

DEFINE ↓

Extend existing mesh → Extend without editing

Mesh number =

After accepting, select **XY plane**, **extrude in Z** from the coordinate system submenu, use the table below to complete the seven remaining extrusions.

Select **Global** coordinate and **Linear** extrusion for each extrusion and then **Accept - and extend again** for the first six extrusions and **Accept - this is the last** for the seventh.

Extrusion Layer	Coordinate	Number of Elements
2	3	15
3	2	10
4	1	10
5	0.5	10
6	0.0	5
7	-5	15
8	-15	5

Press **Return** twice to close all menus.

Defining Materials and Potential Types

To change the material definition from the default settings (**AIR**, **TOTAL SCALAR**) select

MODIFY ↓

Material properties

Layer number	=	1
Accept		Dismiss

and **Accept**, followed by **Select** and **define** in any volume (baseplane facet).

The parameter box should then be completed as follows:

Material Definition		
Material Name	<input type="text" value="air"/>	
Potential Type:	Element Type:	
<input type="checkbox"/> Total Scalar	<input checked="" type="checkbox"/> Linear	
<input checked="" type="checkbox"/> Reduced Scalar	<input type="checkbox"/> Quadratic	
<input type="checkbox"/> Vector		
Options:		
Jx, Jy, Jz	<input type="text"/>	
Vx, Vy, Vz	<input type="text"/>	
Scalar: Charge Density or Rotational Velocity		
Scalar	<input type="text"/>	
Packing factor	<input type="text"/>	
Material orientation		
<input type="checkbox"/> Local XYZ=XYZ	<input type="checkbox"/> Local XYZ=YZX	<input type="checkbox"/> Local XYZ=ZXY
Other vector	<input type="text"/>	
Other volumes and layers:		
From	<input type="text" value="1"/>	To <input type="text" value="*"/> <input checked="" type="checkbox"/> All volumes
Accept	Keep	Help
		Quit

then **Accept**. Press **Finish** to close the **Materials** menu.

Assign the material definitions to the magnet by choosing

MODIFY ↓

Material properties

Layer number	=	3
Accept		Dismiss

and **Accept**, followed by **Select and define**, then select the second facet from the left by clicking near (1.5, 0.5) and complete the parameter box as shown below

Material Definition	
Material Name	magnet
Potential Type:	Element Type:
<input checked="" type="checkbox"/> Total Scalar	<input checked="" type="checkbox"/> Linear
<input type="checkbox"/> Reduced Scalar	<input type="checkbox"/> Quadratic
<input type="checkbox"/> Vector	
Options:	
Jx, Jy, Jz	<input type="text"/>
Vx, Vy, Vz	<input type="text"/>
Scalar: Charge Density or Rotational Velocity	
Scalar	<input type="text"/>
Packing factor	<input type="text"/>
Material orientation	
<input type="checkbox"/> Local XYZ=XYZ	<input type="checkbox"/> Local XYZ=YZX <input type="checkbox"/> Local XYZ=ZXY
Other vector	<input type="text"/>
Other volumes and layers:	
From <input type="text"/>	To <input type="text"/> <input type="checkbox"/> All volumes
Accept	Keep Help Quit

followed by **Accept**.

To modify the first and third facets, pick **Select/deselect volume** and click near (0.5, 0.25), and then pick **Select and define** and click near (2.5, 1.0), and fill in the parameter box as shown below:

Material Definition			
Material Name	<input type="text" value="magnet"/>		
Potential Type:	Element Type:		
<input checked="" type="checkbox"/> Total Scalar	<input checked="" type="checkbox"/> Linear		
<input type="checkbox"/> Reduced Scalar	<input type="checkbox"/> Quadratic		
<input type="checkbox"/> Vector			
Options:			
Jx, Jy, Jz	<input type="text"/>		
Vx, Vy, Vz	<input type="text"/>		
Scalar: Charge Density or Rotational Velocity			
Scalar	<input type="text"/>		
Packing factor	<input type="text"/>		
Material orientation			
<input type="checkbox"/> Local XYZ=XYZ	<input type="checkbox"/> Local XYZ=YZX	<input type="checkbox"/> Local XYZ=ZXY	
Other vector	<input type="text"/>		
Other volumes and layers:			
From	<input type="text" value="3"/>	To	<input type="text" value="4"/> <input type="checkbox"/> All volumes
<input type="button" value="Accept"/>	<input type="button" value="Keep"/>	<input type="button" value="Help"/>	<input type="button" value="Quit"/>

followed by **Accept**.

Press **Finish** to close the **Materials** menu.

Since the induction plate will have eddy currents in it, the vector potential must be used:

MODIFY ↓

Material properties

Layer number	=	6
Accept		Dismiss

and **Accept**.

To select the induction plate consisting of the first, second, third and fourth facets, pick **Select / deselect volume** and click near (0.5, 0.25), (1.5, 0.5) and (2.5,1.0); then pick **Select and define** and click near (6.0, 3.0) and fill in the parameter box as shown below

Material Definition	
Material Name	<input type="text" value="plate"/>
Potential Type:	Element Type:
<input type="checkbox"/> Total Scalar	<input checked="" type="checkbox"/> Linear
<input type="checkbox"/> Reduced Scalar	<input type="checkbox"/> Quadratic
<input checked="" type="checkbox"/> Vector	
Options:	
Jx, Jy, Jz	<input type="text"/>
Vx, Vy, Vz	<input type="text"/>
Scalar: Charge Density or Rotational Velocity	
Scalar	<input type="text"/>
Packing factor	<input type="text"/>
Material orientation	
<input type="checkbox"/> Local XYZ=XYZ	<input type="checkbox"/> Local XYZ=YZX <input type="checkbox"/> Local XYZ=ZXY
Other vector	<input type="text"/>
Other volumes and layers:	
From <input type="text"/>	To <input type="text"/> <input type="checkbox"/> All volumes
Accept	Keep Help Quit

followed by **Accept** and **Finish**.

Defining the Boundary Conditions

Boundary conditions must be applied to the three planes of symmetry as well as the far field boundaries. From inspection, only the tangential components of the fields will be non-zero in the symmetry planes. Therefore, assign **TANGENTIAL MAGNETIC** boundary conditions to these planes.

The far-field boundaries can be set to have either **TANGENTIAL MAGNETIC** or **NORMAL MAGNETIC** boundary conditions (it is assumed that the fields will have died off enough so that the solution should not depend significantly on which conditions are chosen). One side will be set to **NORMAL MAGNETIC** and the rest will be set to **TANGENTIAL MAGNETIC**.

If possible, it is desirable to make at least one surface **NORMAL MAGNETIC** because it allows the software to assign the zero magnetic scalar potential to a surface determined by the user. Otherwise the software arbitrarily chooses a node in the mesh to be the zero potential and this can lead to numerical instabilities in certain situations.

MODIFY ↓

Boundary conditions → Extrusion facets

Layer number	=	1
Accept		Dismiss

and **Accept**. Then select **All external facets** and complete the parameter box:

Global Boundary Condition Definition		
Condition name:		
<input type="checkbox"/> Magnetic Scalar	<input type="checkbox"/> Normal Magnetic	<input checked="" type="checkbox"/> Tangential Magnetic
<input type="checkbox"/> Voltage	<input type="checkbox"/> Normal Electric	<input type="checkbox"/> Tangential Electric
<input type="checkbox"/> Total Ax	<input type="checkbox"/> Total Ay	<input type="checkbox"/> Total Az
<input type="checkbox"/> Incident Ax	<input type="checkbox"/> Incident Ay	<input type="checkbox"/> Incident Az
<input type="checkbox"/> Incident Voltage	<input type="checkbox"/> Perfect Conductor	<input type="checkbox"/> Radiation
<input type="checkbox"/> Normal Derivative	<input type="checkbox"/> Mixed Derivative	
<input type="checkbox"/> Symmetry	<input type="checkbox"/> Slip Surface	<input type="checkbox"/> Clear
Value	<input type="text"/>	Label/2nd Value <input type="text"/>
Accept	Help	Quit

followed by **Accept** and **Finish**.

Finally, reassign the boundary condition of the base plane as **NORMAL MAGNETIC**

MODIFY ↓

Boundary conditions → **Base Plane** → **Select and define**

and select any facet, and complete the parameter box:

Boundary Conditions		
Condition name:		
<input type="checkbox"/> Magnetic Scalar	<input checked="" type="checkbox"/> Normal Magnetic	<input type="checkbox"/> Tangential Magnetic
<input type="checkbox"/> Voltage	<input type="checkbox"/> Normal Electric	<input type="checkbox"/> Tangential Electric
<input type="checkbox"/> Total Ax	<input type="checkbox"/> Total Ay	<input type="checkbox"/> Total Az
<input type="checkbox"/> Incident Ax	<input type="checkbox"/> Incident Ay	<input type="checkbox"/> Incident Az
<input type="checkbox"/> Incident Voltage	<input type="checkbox"/> Perfect Conductor	<input type="checkbox"/> Radiation
<input type="checkbox"/> Normal Derivative	<input type="checkbox"/> Mixed Derivative	
<input type="checkbox"/> Symmetry	<input type="checkbox"/> Slip Surface	<input type="checkbox"/> Clear
Value	<input type="text"/>	Label/2nd value <input type="text"/>
Other volumes and layers:		
From	<input type="text"/>	To <input type="text"/> <input checked="" type="checkbox"/> All facets
<input type="button" value="Accept"/>	<input type="button" value="Keep"/>	<input type="button" value="Help"/> <input type="button" value="Quit"/>

Accept, Finish and **Return** twice to close the modify menu.

Defining the Conductor

Next the conductor is defined.

DEFINE ↓

Conductors → **Define a**

conductor → **Generally**

orientated set → **Racetrack**

The sequence of parameter boxes should be completed as follows:

Local coord 1: X - origin	=	0
Local coord 1: Y - origin	=	0
Local coord 1: Z - origin	=	1.5
<input type="button" value="Accept"/>		

Accept

Local Coordinate system 1	
XYZ local = XYZ global	<input checked="" type="checkbox"/>
XYZ local = YZX global	<input type="checkbox"/>
XYZ local = ZXY global	<input type="checkbox"/>
Other system	<input type="checkbox"/>
Return	←

Return

Local coord 2: X - origin	=	0
Local coord 2: Y - origin	=	0
Local coord 2: Z - origin	=	0
<input type="button" value="Accept"/>		

Accept

Local Coordinate system 2	
XYZ local = XYZ global	<input type="checkbox"/>
XYZ local = YZX global	<input checked="" type="checkbox"/>
XYZ local = ZXY global	<input type="checkbox"/>
Other system	<input type="checkbox"/>
Return	←

Return

Cross-section: width in X	=	0.9
Cross-section: thickness in Y	=	0.9
<input type="button" value="Accept"/>		<input type="button" value="Dismiss"/>

Accept

Bottom inside corner X	=	1.05
Bottom inside corner Y	=	-0.45
<input type="button" value="Accept"/>		<input type="button" value="Dismiss"/>

Accept

Half length of straight	=	0.95
Inside radius of corner	=	0.1
<input type="button" value="Accept"/>		<input type="button" value="Dismiss"/>

Accept

Current density	=	4900
Symmetry code	=	1
Drive label	=	ONE
<input type="button" value="Accept"/>		<input type="button" value="Dismiss"/>

Accept

Conductor reflections	
No reflection in XY(1)	<input checked="" type="checkbox"/>
+ reflection in XY(1)	<input type="checkbox"/>
- reflection in XY(1)	<input type="checkbox"/>
<hr/>	
No reflection in YZ(1)	<input checked="" type="checkbox"/>
+ reflection in YZ(1)	<input type="checkbox"/>
- reflection in YZ(1)	<input type="checkbox"/>
<hr/>	
No reflection in ZX(1)	<input checked="" type="checkbox"/>
+ reflection in ZX(1)	<input type="checkbox"/>
- reflection in ZX(1)	<input type="checkbox"/>
<hr/>	
Return	*

Return

Tolerance on flux density	=	0.01
<input type="button" value="Accept"/>		<input type="button" value="Dismiss"/>

Accept and Return four times.

Creating the Mesh and Displaying the Model

The model will now be meshed with hexahedral elements so that the database may be created.

```
MESH ↓
  Surface mesh ...
    ...quadrilaterals
```

click **Continue** to clear the message box, then pick

```
MESH ↓
  Volume mesh ...
    Mesh
```

and hit **Continue** again.

To display a three-dimensional view of the model select

DISPLAY ↓
Display Command...
...view

and complete the parameter box:

Display View			
Size	<input type="text" value="10"/>		
Eye position:			
X	<input type="text" value="-1"/>	Y	<input type="text" value="1"/>
Z	<input type="text" value="-1"/>		
Centre of picture:			
X	<input type="text" value="0"/>	Y	<input type="text" value="0"/>
Z	<input type="text" value="0"/>		
Rotate picture	<input type="text" value="0"/>		
<input checked="" type="checkbox"/> New picture	<input type="checkbox"/> Add to picture		
<input checked="" type="checkbox"/> Parallel view	<input type="checkbox"/> Perspective view		
<input checked="" type="checkbox"/> Show Axes	<input type="checkbox"/> No axes		
<input type="button" value="Refresh Display"/>	<input type="button" value="Accept"/>	<input type="button" value="Quit"/>	

followed by **Accept**.

Then select

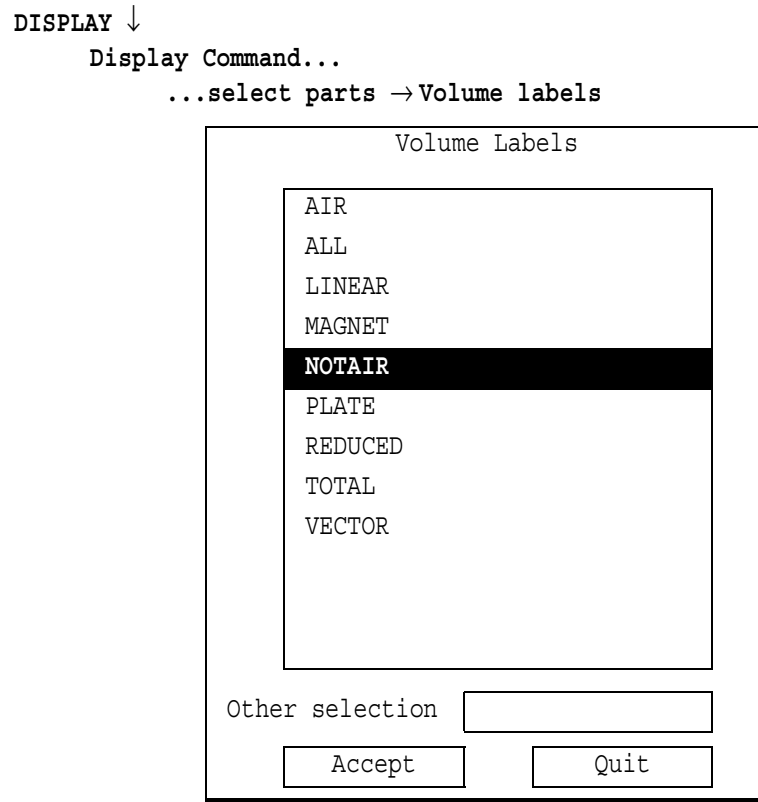
DISPLAY ↓

Display Command...

...style

Display Style	
Line view	<input type="checkbox"/>
Surface view	<input checked="" type="checkbox"/>
Full surface algorithm	<input type="checkbox"/>
<hr/>	
No Elements	<input type="checkbox"/>
Surface Elements	<input checked="" type="checkbox"/>
Volume Elements	<input type="checkbox"/>
<hr/>	
Vectors...	<input type="checkbox"/>
... no vectors	<input checked="" type="checkbox"/>
... in conductors only	<input type="checkbox"/>
... material orientation	<input type="checkbox"/>
... current density	<input type="checkbox"/>
... velocity	<input type="checkbox"/>
<hr/>	
Refresh display	*
<hr/>	
Return	←

and **Return**, followed by:



followed by **Accept**, **Refresh display**, and then **Return** twice.

The resulting display is shown in Figure 10.3

Creating the ELEKTRA Database File

In order to perform a steady-state a.c. analysis on this model, a database must be created.

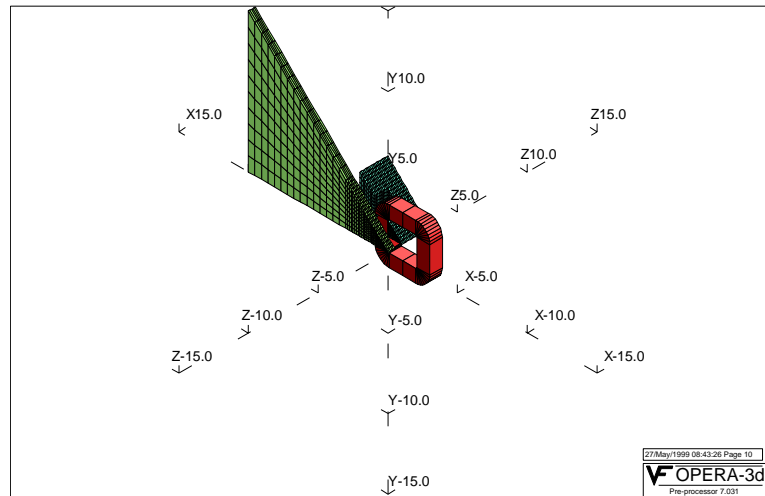


Figure 10.3 Final geometry including conductor

FILE ↓

Analysis

... create new database → Low frequency (ELEKTRA)
Steady-state AC

New Analysis Database File

File

Units: ☒ CGS ☐ SI (metres) ☐ SI (mm) ☐ SI (microns) ☐ Inches

Element type: ☒ Linear ☐ Quadratic ☐ Mixed

and **Accept**.

A box appears prompting for a Problem Title. Type “Levitating Magnet Force calculation” and hit Enter. Next type a single * on its own line and hit enter again. Another box appears asking for the phase angle of the source current, enter 0 and return.

Clear the message box by clicking **Continue**.

The ELEKTRA (steady state) data menu now appears

ELEKTRA (steady state)	
Materials	→
Linear solution	■
Non-linear solution	<input type="checkbox"/>
Adaptive RHS Integrals	<input type="checkbox"/>
Add drive fields	→
Drive frequencies	→
Check data	*
Return	←

Select **Materials** and complete the box as below

Material Names	
MAGNET	Properties of air
PLATE	Properties of air
Material Characteristics	
<input type="checkbox"/> Linear	<input checked="" type="checkbox"/> Non-linear
<input checked="" type="checkbox"/> Isotropic	<input type="checkbox"/> Packed
<input type="checkbox"/> Multiple	
<input type="button" value="Define"/>	<input type="button" value="Return"/>

Select **Define** and complete the following box.

Properties for material MAGNET

Non-linear BH curve

Isotropic

Complex phase lag

Linear Conductivity [SCM]

Isotropic

Complex phase lag

and **Accept**. If the file with the default BH curve is not in your working directory, you will have to provide a path to that file. You can find the file in the BH subdirectory of your OPERA installation on your hard disk. In the **Selected Directory** section, type one of the following pathnames.

UNIX operating systems, use: `$vfdir/bh`

and on Windows use: `%vfdir%\bh`

Hitting **Return** (or **CD**) will change to that directory, and will display the available BH curves, including *default.bh*.

The plate will be defined in a similar manner except that it will be treated as a linear material

Material Names

MAGNET	Non-linear, isotropic
PLATE	Properties of air

Material Characteristics

☒ Linear
 ☐ Non-linear

☒ Isotropic
 ☐ Packed
 ☐ Multiple

Again select **Define**

Properties for material PLATE	
Linear permeability	
Isotropic	<input type="text" value="1"/>
Complex phase lag	<input type="text" value="0"/>
Linear Conductivity [SCM]	
Isotropic	<input type="text" value="6e5"/>
Complex phase lag	<input type="text" value="0"/>
<input type="button" value="Accept"/>	<input type="button" value="Quit"/>

and **Accept**, followed by **Return**.

Select **Non-linear solution** and complete:

ELEKTRA nonlinear iteration data	
<input type="checkbox"/>	<i>Newton-Raphson</i>
<input type="checkbox"/>	<i>Simple Update</i>
Number of iterations	<input type="text" value="15"/>
Convergence tolerance	<input type="text" value="0.001"/>
Underrelaxation factor	<input type="text" value="1"/>
<i>NL Iterations per timestep</i>	<input type="text"/>
<input type="button" value="Accept"/>	<input type="button" value="Dismiss"/>

and **Accept**.

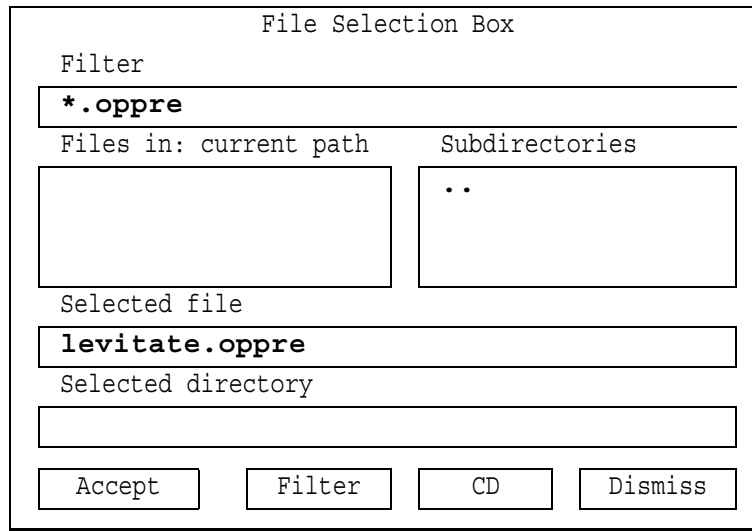
Select **Drive frequencies** and enter **60** for the new frequency. Press **Add** to add this frequency to the case list, and close the menu with **Return**.

Close the ELEKTRA (steady state) data menu with **Return**, then clear the message box by clicking **Continue**.

Write out a pre processor file which will contain the model along with the analysis options chosen so far.

FILE ↓

Write pre-processor file



The image shows a 'File Selection Box' dialog. It has a title bar 'File Selection Box'. Inside, there is a 'Filter' section with a text box containing '*.oppre'. Below this, there are two columns: 'Files in: current path' and 'Subdirectories'. The 'Files in: current path' column contains a list box with 'levitate.oppre' selected. The 'Subdirectories' column contains a list box with '..'. Below these columns, there are two more text boxes: 'Selected file' (containing 'levitate.oppre') and 'Selected directory' (empty). At the bottom, there are four buttons: 'Accept', 'Filter', 'CD', and 'Dismiss'.

and **Accept**. Click on **Continue** to clear the message box and then **End OPERA-3d/Pre** to exit the pre processor.

The model must then be submitted to ELEKTRA-SS for analysis.

Note that although it is possible to launch the analysis directly from the pre processor, this does require extra system resources. It may be necessary to close the pre processor and run the analysis separately (see following section).

Running the ELEKTRA Simulation

UNIX Operating Systems

From within the OPERA-3d environment select the ELEKTRA option

Option:

elektra

and from the following prompt:

Please give ELEKTRA database filename (without the .OP3 suffix)

levitate

Indicate which type of ELEKTRA solution is required (SS in this case)

Steady state AC (ss), Transient (tr) or Velocity (vl)?

ss

and choose that the analysis is carried out immediately:

Do you want to run the analysis now or later? (n or l)

n

The analysis then proceeds automatically.


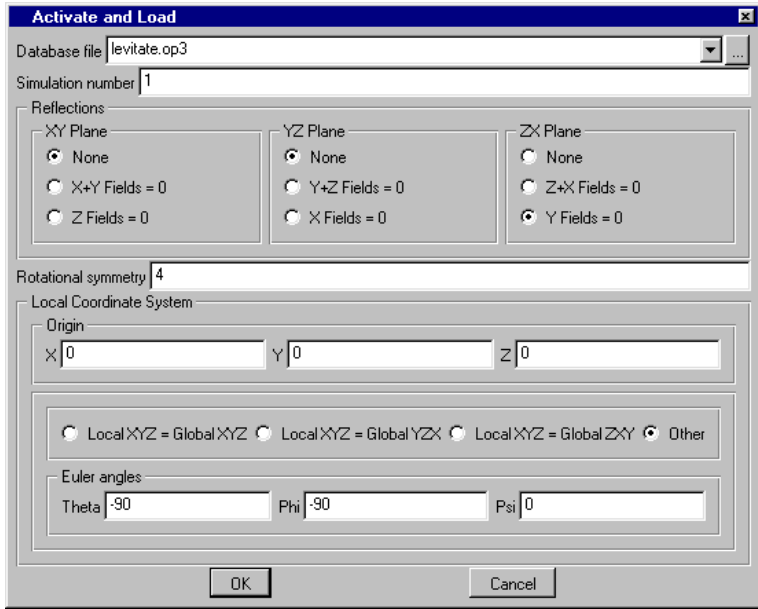
Windows Systems

Choose interactive solution under the OPERA-3d menu in the OPERA Console. Select the ELEKTRA-SS solver and browse to where the *levitate.op3* file was saved and select it as well. The analysis module will then proceed with the calculations.


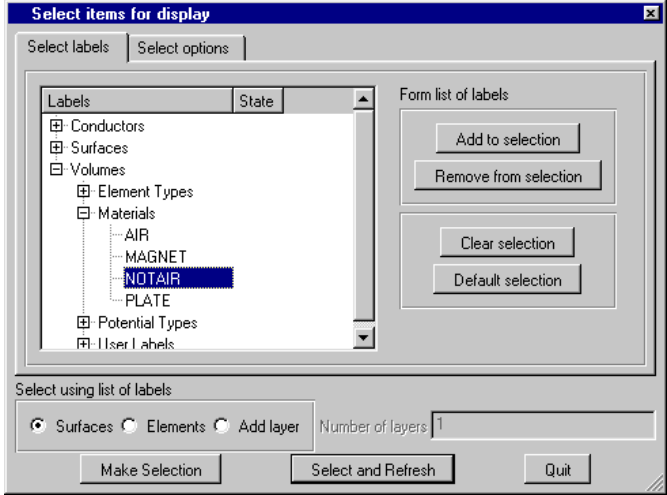
The pre processor data file (*.oppre*) is also provided with the OPERA installation (see the sub-folder *Examples/3D*) if required.

Analysing the results


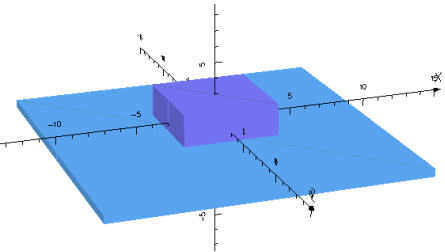

Launch the post processor from the OPERA console and read the solution file with the following settings:

<p>Select the Load file icon</p> 	<p>In particular, note the new local coordinate system that is used in displaying this model.</p> 
---------------------------------------------------------------------------------------------------------------------------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Clicking on the **OK** button will confirm the chosen settings. To display the model select,

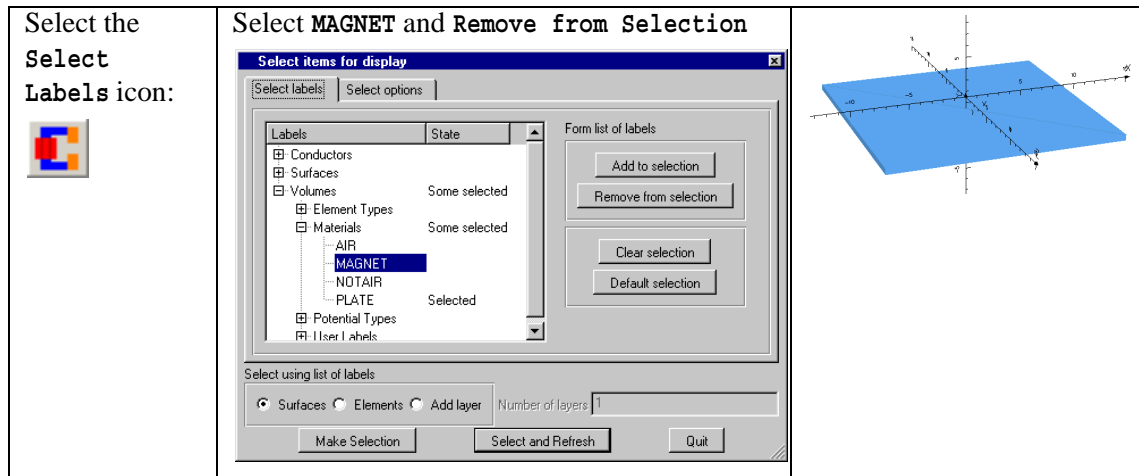
<p>Select the Select Labels icon</p> 	<p>Click on Volumes, followed by Materials in order to highlight NOTAIR</p> 
-------------------------------------------------------------------------------------------------------------------------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Having selected **NOTAIR** from the **Volumes** list, click on **Add Selection**, followed by **Make Selection**.

<p>Select Initial View icon</p> 	<p>In order to display the selection.</p>	
<p>Select Outline icon:</p> 	<p>Switch off the outline view of the model, to allow a clear display of flux density and eddy currents.</p>	

Displaying the flux density and eddy currents

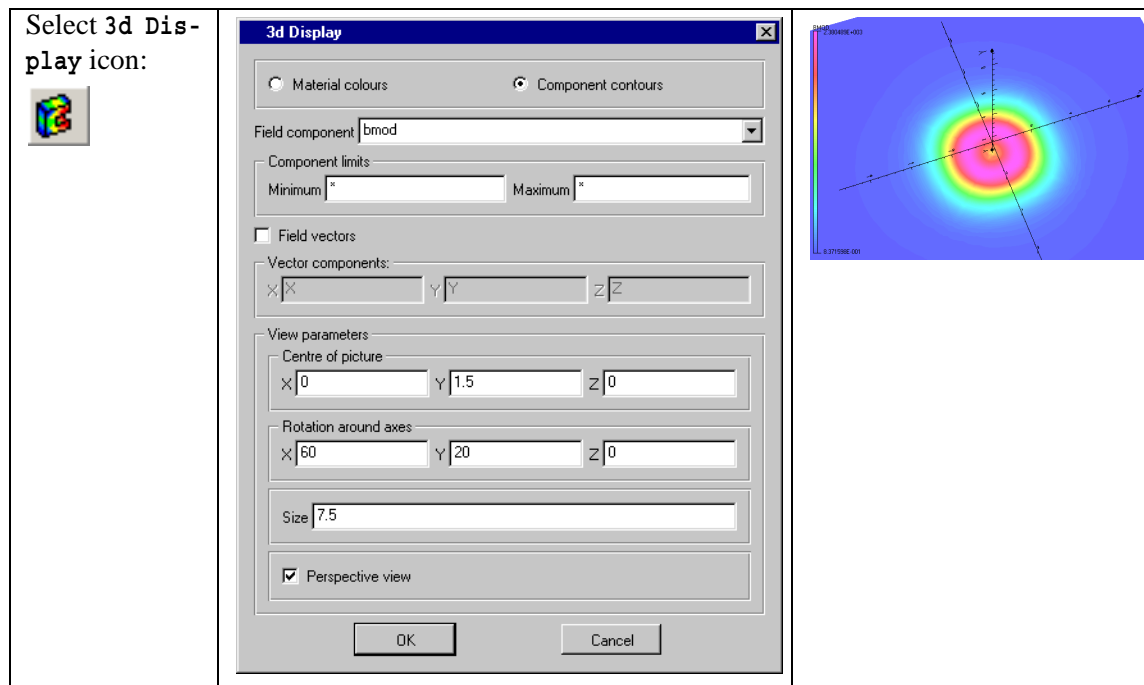
De-select the magnet as follows:



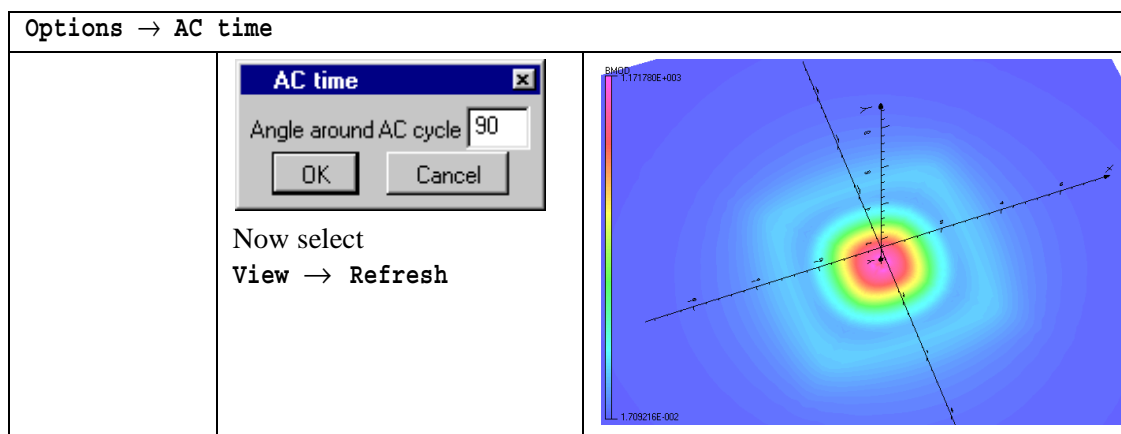
Now click on **Select and Refresh**.

Finally, the field component must be selected for display, and the size of the display adjusted for clarity.

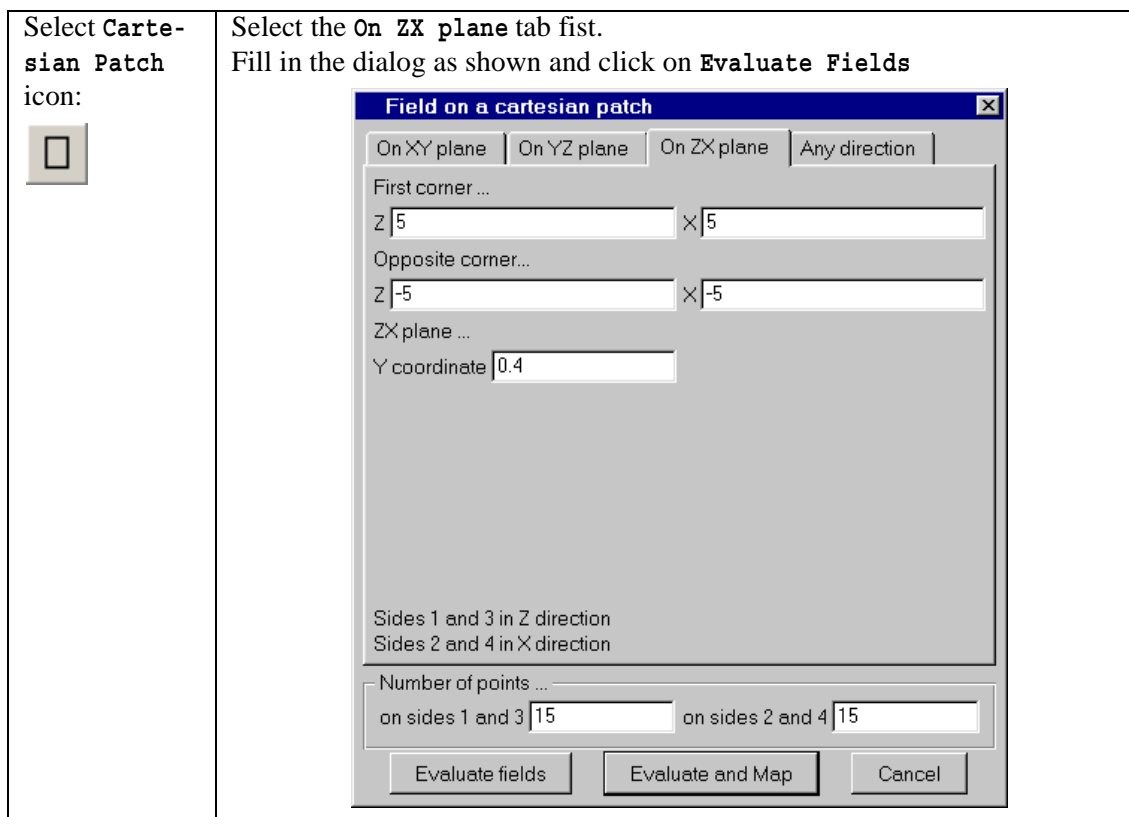
Fill in the 3D Display Menu as shown below to obtain the accompanying plot.

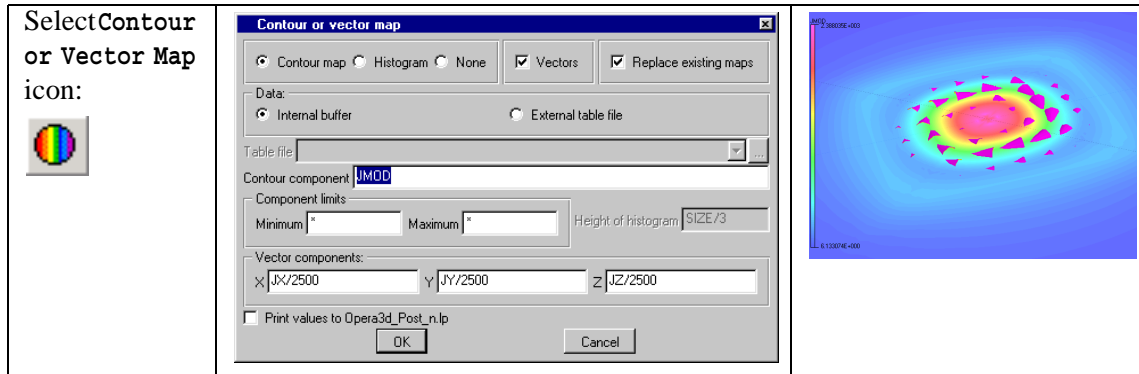


The flux density distribution corresponds to the peak source current density ($t=0$) and has a maximum value of $2.38 \text{ E}+03$ Gauss. We can also plot the quadrature component by choosing



To display the eddy current density, including the direction at $t=90$, a cartesian patch may first be defined.

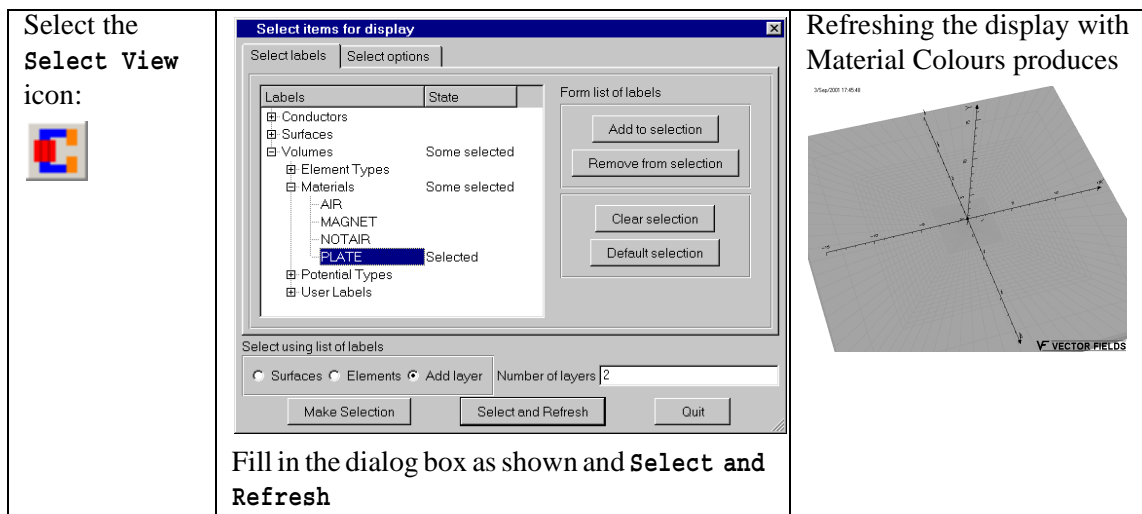




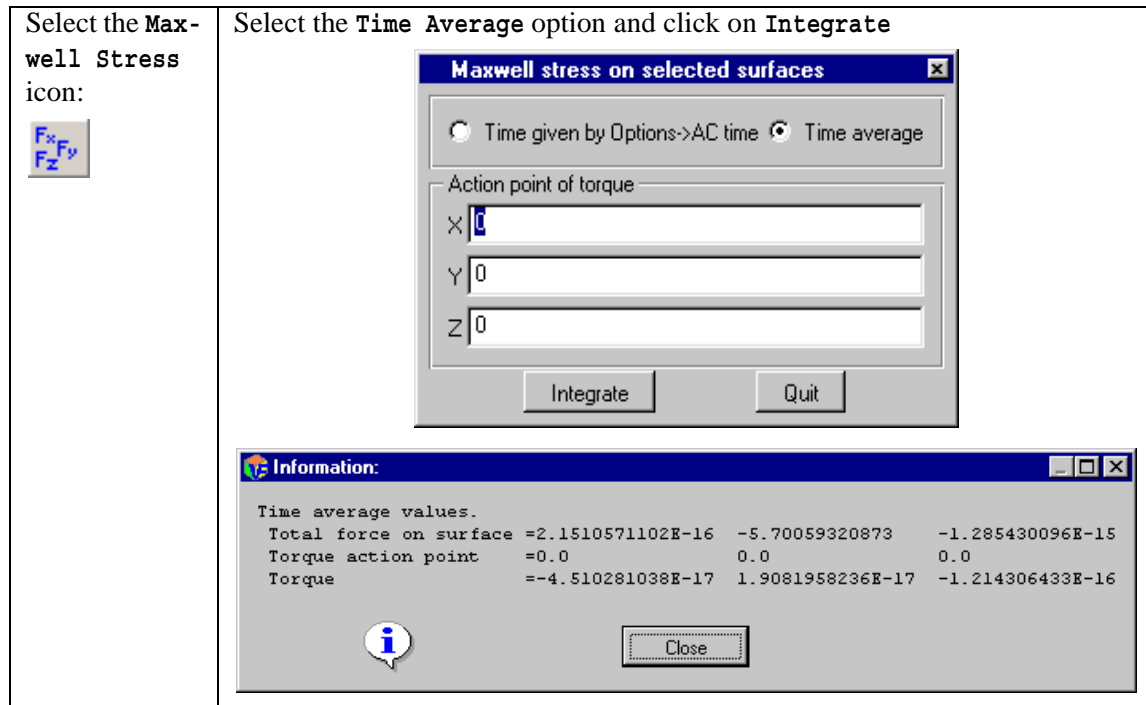
This yields a maximum current density of $2.388 \text{ E}+03 \text{ A/cm}^2$.

Calculating the levitation force

The repulsive force between the magnet and plate can be calculated by integrating the Maxwell stress over a surface which surrounds the plate. Two layers of air elements will be added to the plate before performing the integral



The integral may now be performed.



The reason for adding two layers of elements around the plate is that the integration is more accurate if performed over a surface which is not in direct contact with the body of interest. The result indicates that the levitation force on the magnet is 5.7 Newtons and is directed upward as expected.

Chapter 11

CARMEN Example

Introduction

CARMEN is part of the OPERA-3d suite of software and is the 3 dimensional rotating machine analysis module. It computes time varying fields and resulting eddy currents in rotating structures. Figure 11.1 shows a typical post processed CARMEN analysis (stator coils omitted).

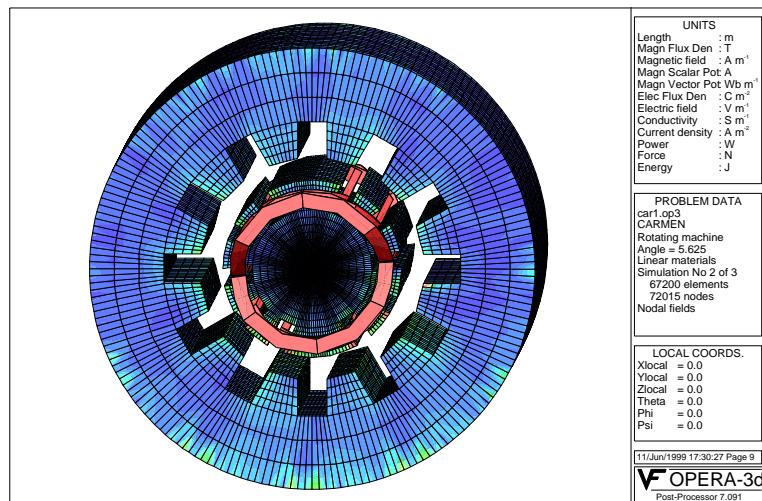


Figure 11.1 A CARMEN model

The rotating part of the model (the rotor) is enclosed by the fixed part of the model (the stator). The main difference between an ELEKTRA model and a CARMEN model is that for CARMEN the user must create a slip surface in the air gap which defines the extent of the moving part of the geometry.

The mesh along the slip surface must be uniform and evenly spaced so that as the rotor moves the mesh is always continuous across the surface, see Figure 11.2. A consequence of this is that hexahedral elements must be used for CARMEN models.

The mesh created along the slip surface directly defines the angle of each rotation of the rotor. For example if there are 360 subdivisions around the complete slip surface the rotor will move in increments of 1 degree.

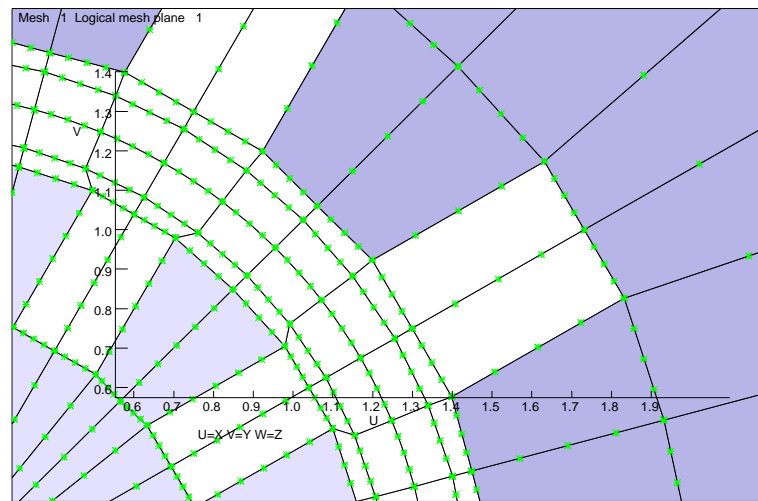


Figure 11.2 The CARMEN slip surface

The nature of this ‘lock-step’ algorithm means that apart from a possible central symmetry plane, the complete model must be created. This can lead to large models and consideration of this must be made when using the software.

If users have other OPERA software besides CARMEN, it is recommended that this is NOT the first tutorial run. Other tutorials give more information on the use of the pre processor to set up models and guide users through the stages of creating a model. For users who have CARMEN only, previous tutorials may also prove valuable to gain familiarity with the pre processor.

Features included in this tutorial are:

- Modelling of a synchronous generator.
- Partially complete *oppre* file included on the distribution-CD.
- Definition of slip surface.
- Setting up the database for analysis including a 3 phase coil system in the stator and a dc coil in the rotor.

- Using the post processor to extract results.

This tutorial represents a realistic example of a generator problem. It is assumed that the output current is known and this will be applied to the stator coils. In this case there are 2 coils per slot and the current is applied in three phases. In this model, long straight conductor bars have been used in the stator as end effects were not considered to be important. The ends of the conductors have been placed at some distance from the model so as not to influence the result. Remember that with OPERA-3d, the coils are not part of the mesh and can extend beyond the user defined mesh if required. CARMEN is a transient code and it is possible that switch-on transients from sinusoidal or cosinusoidal excitations may influence the results for the initial steps of the rotation.

Loading the CARMEN Data File

Launch the pre processor in the normal way, and read the *.oppre* file provided on the installation CD.

The pre processor data file (*.oppre*) is provided within the OPERA installation (see the sub-folder *Examples/3D*). This file has the region names set and the subdivisions specified. It has been created by defining a repeating segment of the generator geometry and then using **GROUP OPERATIONS** to create the complete model. The repeating segment should be visible when the *.oppre* file is first loaded. There are 2 layers in the model and a symmetry plane is defined so that only half the motor (in the axial direction) is defined.

Once the *.oppre* file is loaded, have a look at the baseplane to gain an idea of the complexity of the model. This is carried out as follows:

MODIFY ↓

Materials Properties

Select plane number 1 and press **Accept**

Resize the display if necessary by:

Redraw picture → **Bounding rectangle**

The complete base plane is now visible and the material boundaries of the rotor and stator should be discernable.

Resize the display again by

Redraw picture → **Numerical limits**

and complete the dialog as shown below:

Minimum on horizontal axis	=	0
Maximum on horizontal axis	=	2
Minimum on vertical axis	=	0
Maximum on vertical axis	=	2
<input type="button" value="Accept"/> <input type="button" value="Dismiss"/>		

More detail of the generator can now be seen including the structure of the air gap between the rotor and stator. The slip surface itself is not yet defined, but will shortly be set to the centre line of the 4 facets across the airgap.

Note that it is important to have a minimum of 4 facets radially across the air gap. This is because the air regions abutting the slip surface must be **TOTAL POTENTIAL**. The air regions beyond these should be set to **REDUCED POTENTIAL** to avoid loops of total potential enclosing a coil in a slot, thus avoiding a multiply connected region (see Figure 11.3).

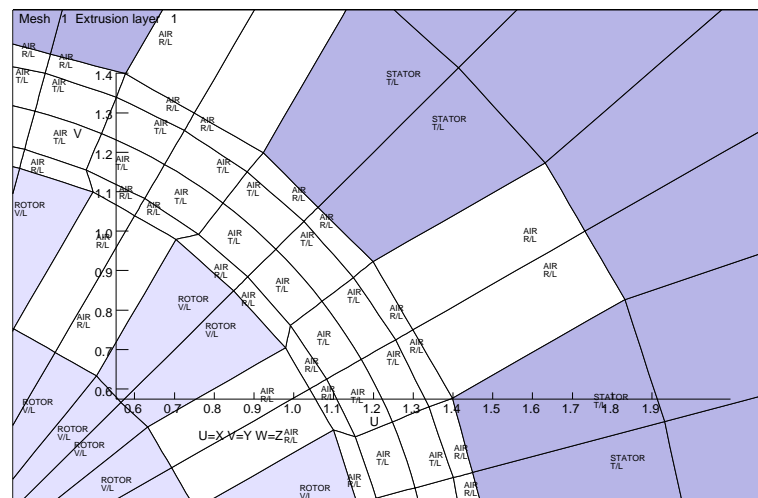


Figure 11.3 Facets in Air-Gap

Display the region potential by selecting

Show Properties

Select:

Finish

and select from the menu:

Subdivisions → Uniform subdivision → In plane

The previous picture will be redrawn showing the green subdivision markers. These are used by the meshing software to create the finite element mesh. Note that the subdivision along the edge where the slip surface is to be placed (at radius 1.35), is uniform. The space between each marker is identical. Also note how the facet edges on this surface have been made the same length. This is not essential, but ensures that if the same subdivision is applied to each facet edge the spacing will be the same. In this case a subdivision of 4 was applied to each facet edge.

It is essential for the correct functioning of the CARMEN analysis module that the subdivision spacing along the slip surface is uniform.

Adding the SLIP Surface

To add a SLIP surface to the model, select

Finish... **Valid Hex Mesh?** and **Return** twice,

then select

MODIFY ↓

Add slip surface

Enter the radius of the slip surface in the dialog (this will be 1.35), and leave the default tolerance on the radius to be 0.001. Select **Accept**, and a message appears to indicate that the slip surface boundary has been added to 96 facets (that is all facets on this radius, for the entire length of the model).

Note that the subdivision on each slip surface facet was set to 4 giving a total of 192 subdivisions around the complete slip surface. The rotor will therefore rotate in increments of 1.875 degrees. Note also that the slip surface must be at the same radius for every layer of the model

The model is now complete and almost ready for analysis. Before that can happen, a mesh and analysis database need to be created

Creating the mesh

The model has been created with 4 sided facets on the base plane and can therefore be meshed with the hexahedral mesh generator. First of all create the surface mesh:

```
MESH ↓
    Surface mesh...
        ...quadrilaterals
```

Then create the volume mesh with

```
Volume mesh...
    Mesh
```

Return from the menu and select:

```
DISPLAY ↓
    Refresh display
```

An end-on view of the machine should be visible along with the surface mesh and the coils. It can be seen that the inner rotor coils form loops, but that the outer coils are simply straight bars. Although this is clearly non-physical, the correct current will be applied by the software.

Coil Definition

The coils have been pre-defined for you, but it is instructive to see how they have been created.

Return from the **DISPLAY** menu.

Select:

```
DEFINE ↓
    Conductors → Print data
```

In the dialog box enter first and last as **1** and **Accept**

A display panel will appear with all the details of conductor 1. Remember that until the analysis database is created, the length figures shown are unitless.

- It is a straight bar conductor from the OPERA-3d library.
- The local coordinate system 1 is the same as the global coordinate system (0,0,0). There is no rotation of one with respect to the other.
- The local coordinate system 2 is shifted so that the centre of the end face of the conductor is at (1.6,0,-100).
- The conductor has a square cross section with a side length of 0.1.
- The length of the bar is 200.5 so that the bar extends equally in the positive and negative axial direction.

- There is no symmetry or reflections applied.
- The phase parameter is a *label* and in this case it is called **1**. (It could equally have been **ONE** or any other name). The current density is applied to the coil. If a sinusoidal variation is applied then this is the peak value of the sine wave.

The first conductor in this model is the inner conductor in the 0 degree slot. See

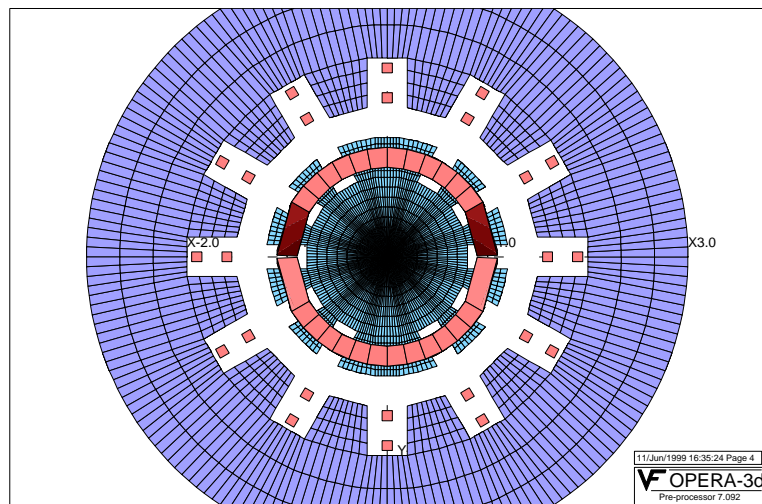


Figure 11.4 Conductor Layout

Figure 11.4 for a diagram showing the conductors. Conductors 1 to 24 are the stator bars. Conductors 25 to 27 are the rotor coils and are defined as helical end racetracks from the conductor library.

Have a look at the information for some of the other conductors. In particular notice how the phase label changes on the stator windings. This is a 2-pole three phase generator and the phases are arranged so as to give as smooth a 3 phase output as possible. With this winding arrangement there are effectively 2 magnetic poles. The dc windings on the rotor have been set up to give a 2 pole field and this can be seen by selecting conductors 25 to 27 and examining the direction of the currents in the windings. The generator is therefore set up to produce one cycle of ac for one revolution of the rotor.

Viewing the geometry

Return to the main menus and select:

DISPLAY ↓

Display Command...

... select parts

and press the **Conductors** button to toggle it to **No conductors**.

Press **Return**.

DISPLAY ↓

Display Command...

...view

and fill in the dialog box as below:

Display View			
Size	3		
Eye position:			
X	1	Y	1
Z	3		
Centre of picture:			
X	0	Y	0
Z	0		
Rotate picture	0		
<input checked="" type="checkbox"/> New picture	<input type="checkbox"/> Add to picture		
<input checked="" type="checkbox"/> Parallel view	<input type="checkbox"/> Perspective view		
<input checked="" type="checkbox"/> Show Axes	<input type="checkbox"/> No axes		
Refresh Display		Accept	
		Quit	

Press **Accept** and **Refresh** the display. A 3-dimensional view of the machine will be displayed.

Return to the main menu and select:

DISPLAY ↓

Display Command...

... select parts → **Facet labels**

and in the selection box select **SLIP**. Press **Accept** and subsequently **Refresh Display**. The slip surface should be shown in green.

Creating the Analysis Database

The final pre processing stage is to create the analysis database. This should be a familiar operation to users who have tried the other tutorials or who have used OPERA-3d before.

Go to the top menu bar and select the following:

```
FILE ↓
    Analysis
        ...Create new database
```

If **Create new database** is not available (it is low-lit), this indicates that the mesh has not been created. Return to “creating the mesh” section above.

Otherwise from the selection given choose

Rotating Machine (CARMEN)

Fill in the dialog box as shown below:

The dialog box is titled "New Analysis Database File". It contains the following fields and options:

- File:** A text box containing "carmen" and a dropdown arrow button.
- Units:** A group of radio buttons:
 - ☐ CGS
 - ☒ SI (metres)
 - ☐ SI (mm)
 - ☐ SI (micros)
 - ☐ Inches
- Element type:** A group of radio buttons:
 - ☒ Linear
 - ☐ Quadratic
 - ☐ Mixed
- Buttons:** "Accept" and "Quit" buttons at the bottom.

Press **Accept**.

OPERA will now create the binary database on disk. This may take a couple of minutes. A dialog box will appear asking for a name for the problem you are creating. Enter the text "**CARMEN Generator**" (without the quotes) followed by return. To complete the title enter * and return

Entering the Coil Drive Data

A large dialog box will appear requesting drive data for coil drive 1, label 1. In the check box select **COSINE**. Fill in the Frequency as **50**. Fill in the phase as **0**. Press **Accept**

A further dialog box will appear requesting drive data for coil drive 2, label 2. In the check box select **COSINE**. Fill in the Frequency as **50**. Fill in the phase as **120**. Press **Accept**

A third dialog box will appear requesting drive data for coil drive 3, label 3. In the check box select **COSINE**. Fill in the Frequency as **50**. Fill in the phase as **240**. Press **Accept**

A final dialog box will appear requesting drive data for coil drive 4, label **ROTOR**. In the check box select **DC** and **ROTATING** Press **Accept**.

*It is very important to make sure that the **ROTATING** box is checked. If not the rotor coil will remain stationary as the rotor moves which will give incorrect results.*

The CARMEN data menu now appears:

CARMEN	
Materials	→
Linear solution	<input checked="" type="checkbox"/>
Non-linear solution	<input type="checkbox"/>
Adaptive RHS Integrals	<input type="checkbox"/>
Add drive fields	→
Rotation speed	→
Output angles	→
Check data	*
Return	←

Entering the Material Properties

Select **Materials** and the following window should appear.

Material Names	
STATOR	Properties of air
ROTOR	Properties of air

Material Characteristics

☒ Linear
 ☐ Non-linear

☒ Isotropic
 ☐ Packed
 ☐ Multiple

The line which says “STATOR Properties of air” should be highlighted. Leave the check boxes set at linear isotropic. Click on **Define**. If the line is not highlighted then click on the line with the mouse. A dialog box should appear with the title “Properties of material STATOR”. Fill in the table as below and then press **Accept**:

Properties for material STATOR		
	Linear Permeability	Coercive Force [AM]
Isotropic	<input type="text" value="100"/>	<input type="text" value="0"/>
	Linear Conductivity [SM]	
Isotropic	<input type="text" value="0"/>	
<input type="button" value="Accept"/>		<input type="button" value="Quit"/>

Next highlight the line from the menu “ROTOR Properties of air” and again fill in the dialog box as below:

Properties for material ROTOR		
	Linear Permeability	Coercive Force [AM]
Isotropic	<input type="text" value="20"/>	<input type="text" value="0"/>
	Linear Conductivity [SM]	
Isotropic	<input type="text" value="3e6"/>	
<input type="button" value="Accept"/>		<input type="button" value="Quit"/>

Click **Accept** and press **Return**.

Rotation Speed

Select **Rotation speed** from the CARMEN data menu.

A small dialog box will appear requesting the rpm (revolutions per minute). As this is a synchronous machine running with a coil frequency of 50 Hz, the speed of the rotor needs to be set at 3000 rpm. Enter **3000** in the dialog box and return. Ignore the section on time step (this only important if the rotation speed is zero).

Output Angles

Select **Output angles** from the CARMEN data menu.

This dialog box will request **Output angles**. The software will not output data for every angle solved as this would produce an overwhelming amount of data. Instead the user is requested to enter angles at which data would be useful. Enter the following angles by pressing the **Add** button each time

5, 20, 90.

We have requested output data at 5, 20, and 90 degrees. 0 degrees is automatically included. Therefore this analysis will produce four cases 1, 2, 3, 4 corresponding respectively to angles 0, 5, 20 and 90 degrees. The case is selectable in the post processor as will be seen shortly.

Note that the nearest angle to that requested will be used. Remember that due to the subdivisions specified, the increment is 1.875 degrees. The table below shows the first 14 angles and the corresponding cases.

0	→	case 1
1.875		
3.75		
5.625	→	case 2
7.5		
9.375		
11.25		
13.125		
15		
16.875		
18.75		
20.625	→	case 3
22.5		
24.375		

Important notes:

- Although we are entering a sinusoidally varying excitation, CARMEN runs as a transient code. This means that there will be ‘switch-on’ effects. To eliminate these it would be necessary to run for more steps. Angles where identical results are expected can be compared to check for switch-on effects. Clearly this would entail a longer analysis and is omitted here to save time.
- Another option would be to run the problem at low speed initially and subsequently restart with the actual (higher) speed and/or different properties. Note that restarts can only be performed from data files analysed with the same solver.

Enter **Return** to close the list of the output angles.

Save the Data File

The database is now complete and has been written to disk. Press **Return** to close the CARMEN data menu. A summary of the database will be shown on the screen. Click **Continue**.

Save the modified *.oppre* file to disk. This will also save the pre processing you have carried out to create the database, appended as comments to the end of the original file. Close the pre processor.

OPERA-3d Analysis

Start the analysis using

FILE ↓

Start analysis now

and note that the required items are already selected.

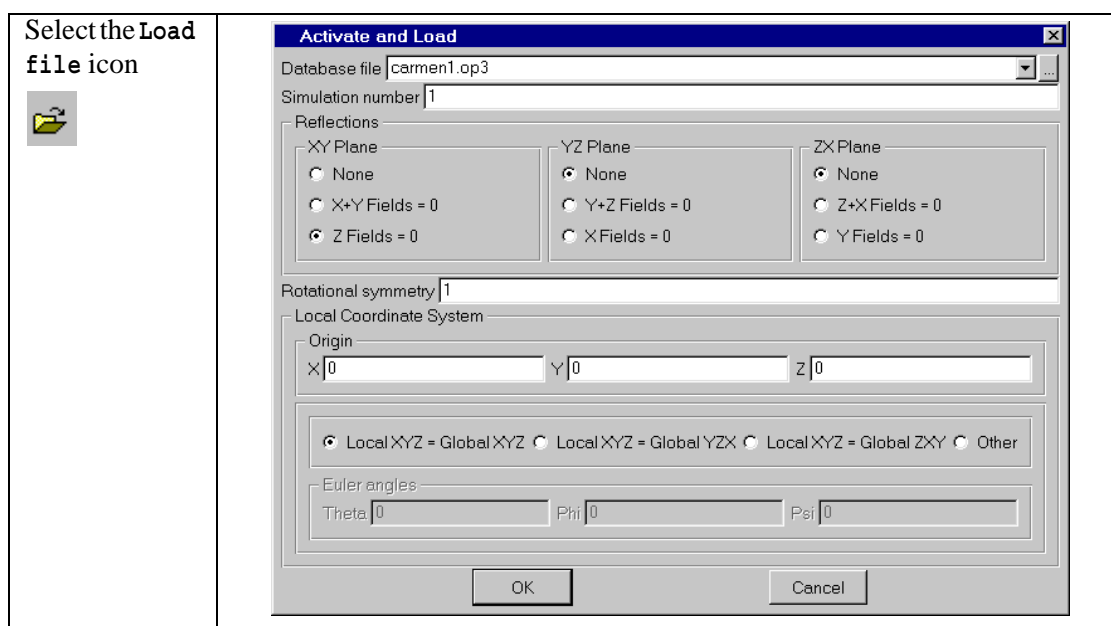
Warning: This analysis represents a real rotating machine. It will take about 8 hours to solve on a 1 GHz PC (about 10 minutes per time step). It is recommended that it is run overnight or on an unused machine. The actual size of this model is a database with about 153000 equations.

OPERA-3d Post Processor

Loading the CARMEN solution

The OPERA 3d model created was for only half of the generator. The rest of the model was implied by appropriate boundary conditions, in this case tangential magnetic at the reflection interface. At the loading stage we can apply appropriate reflections to see the whole model.

Launch the post processor and read the solution file with the following settings:



Clicking on the **OK** button will confirm the chosen settings.

The simulation number (or case number) is set at 1. This is always the zero degree case. For the database we created, case 2 is 5 degrees, case 3 is 20 degrees, case 4 is 90 degrees and case 5 is 180 degrees (all approximated to the nearest time step).

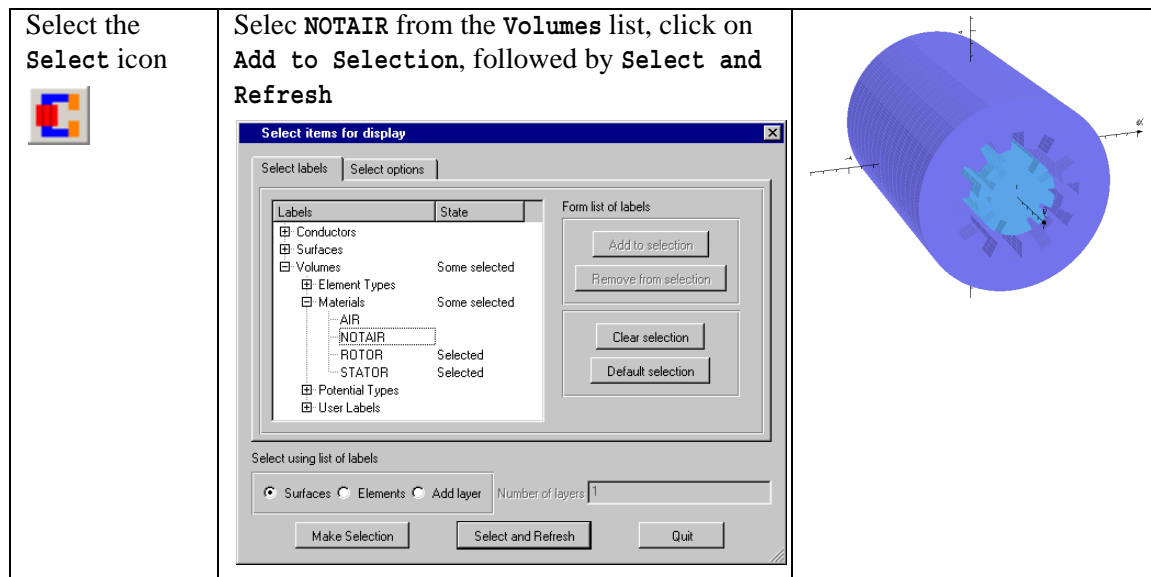
Under **Reflections**: the **XY Plane, Z Field=0** box was checked, ensuring that the post processor applies the correct boundary condition at the interface.

Viewing Parts

There are two stages to viewing a model in the post processor. First of all the parts to be viewed must be selected and subsequently the options for viewing may be set.

Selecting Parts and Displaying

In order to select the appropriate components for display,




The **Select and Refresh** option allows for the simultaneous execution of the two steps to viewing the model (that is, selection and display as described earlier). The software defaults to pre-set viewing options to achieve this.

Viewing Fields

There are many ways of viewing data in the post processor. Initially we will look at the surface shading of a field component. Any field component in the analysis can be used for this purpose.

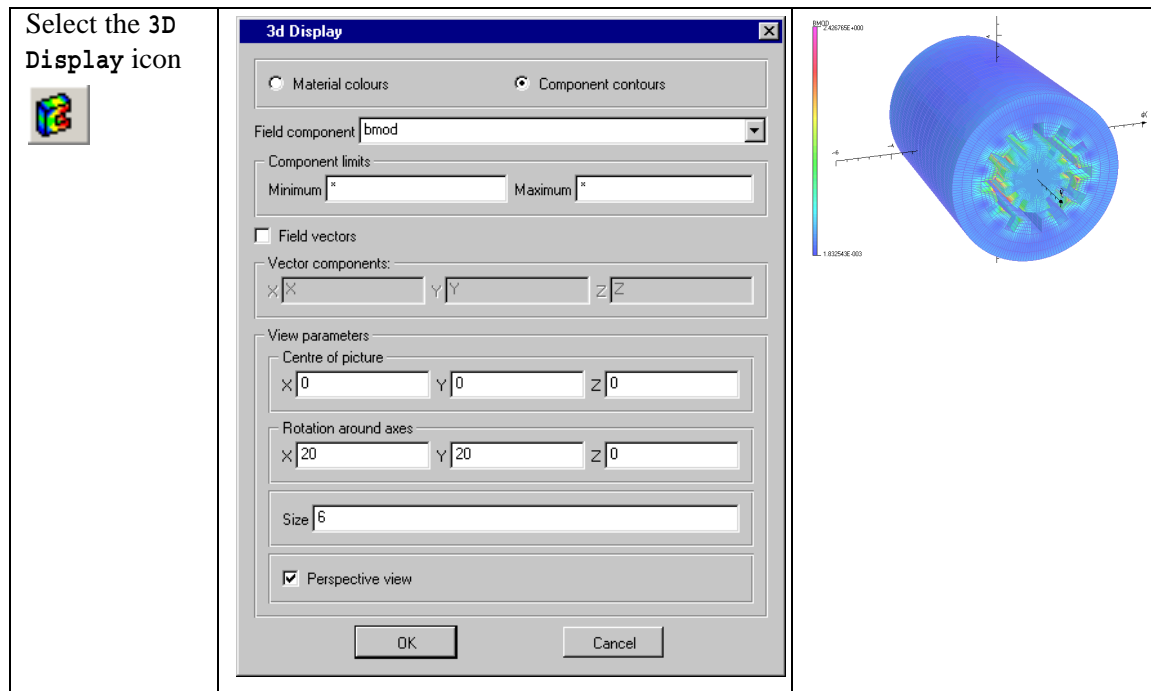
Surface Fields

In order to display the flux density distribution in the machine, the outline view of the model ought to be switched off to allow a clear view of the field colour con-


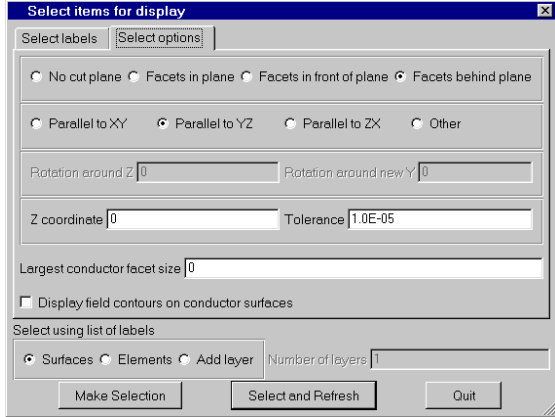
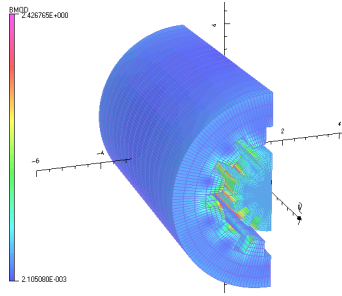
tours. Click on  or alternatively follow menu route

View → Parts of the display → Outline view of model (toggle)

Subsequently, the field component must be selected for display and the size of the picture adjusted. Fill in the dialog as shown below to obtain the accompanying plot.



In CARMEN, the complete model must be generated in the direction of rotation. To see features and fields on inside surfaces, the ‘cut plane’ menu may be used.

<p>Select the Select icon:</p> 	<p>Click on Select options tab and select facets behind the YZ plane:</p>  <p>Click on Select and Refresh.</p>	
-------------------------------------------------------------------------------------------------------------------------	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-------------------------------------------------------------------------------------

The picture displayed will show a cut-away of the model. Cut planes at other angles can be specified although the cut will not divide facets, so the result may have jagged edges.

If fields on a plane are required, a better option is to use the **Cartesian Patch**

icon: , or

Fields ↓

Fields on a cartesian patch

Try the previous section using **JMOD** instead of **BMOD**. You should find that there is no induced current. This is correct behaviour as this is the initial state with no eddy currents.


Try going back to the beginning of the post processing section, (“**OPERA-3d Post Processor**” on page 11-15), selecting Case 2 (5 degrees) and following the same procedure. This time there should be some eddy currents apparent on the surface. Fields and currents for other cases may also be examined.

Motor Performance

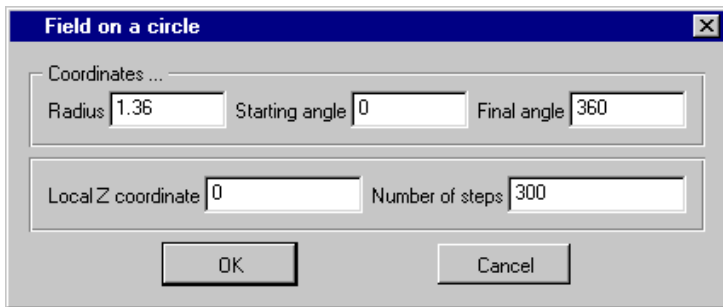
It is useful to have a look at the radial flux in the air gap of the machine so that we can see how the motor or generator is performing. This is a 3-phase, 3000 rpm

generator running at 50 Hz. For each complete rotation of the rotor, one cycle of ac should be generated. Load Case 1 (0 degree position).


Select **Fields** on a circle icon:

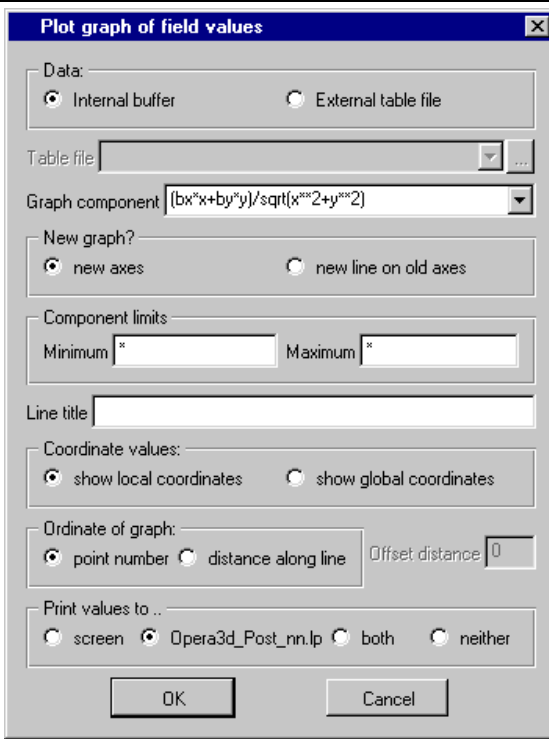


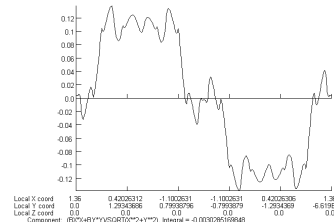
Fill in the dialog box as shown and click on **OK**



Select **Plot** **Graph of field values** icon






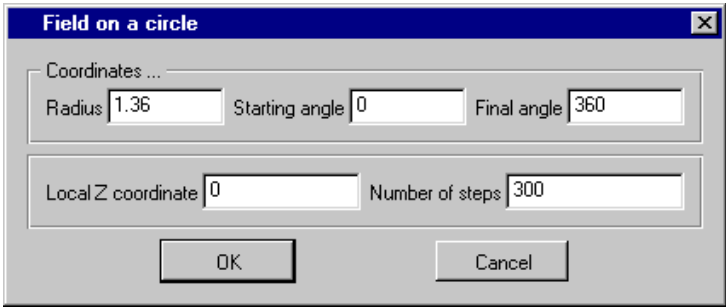

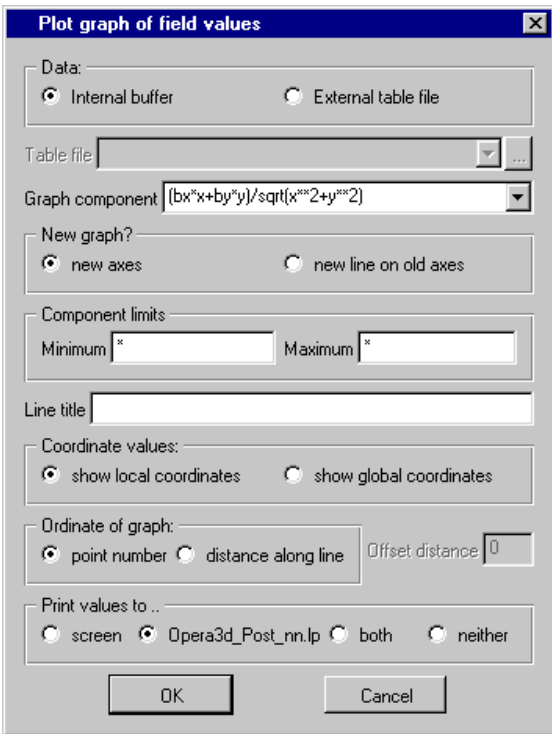
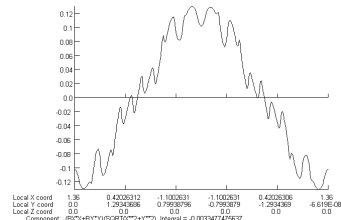


The component specified is the radial flux density. In 3D, the software has the normal vector to a surface specified as the system variables, **NX**, **NY** and **NZ**. This cannot be used in this case as we are looking at fields on a line where in 3D, there are an infinite number of normals to each point on the line.

The above expression is equivalent to $BX \cdot \cos(\theta) + BY \cdot \cos(\theta)$ where θ is the angle from the point on the circle to the local x axis.

The graph correctly shows one ac cycle. The 'spikes' on the graph are due to the stator slotting.

Load in Case 4 (90 degrees) and repeat the above process. .

<p>Select Fields on a circle icon:</p> 	<p>Fill in the dialog box as shown and click on OK</p> 
<p>Select Plot Graph of field values icon</p> 	 

The graph still shows one cycle of ac but this time the plot is phase shifted by 90 degrees.

Chapter 12

A Space Charge Example

Introduction

In this example we will consider a high-voltage electron gun with a curved emitting surface. A cross-sectional view along the length of the gun is shown in Figure 12.1. The features demonstrated in this example include: exploiting symmetry, quadratic extrusions, point editing, and the creation of emit files.

The device is modelled using hexahedral elements. This is not essential but tight control of the mesh close to the emitter is important for accurate results.

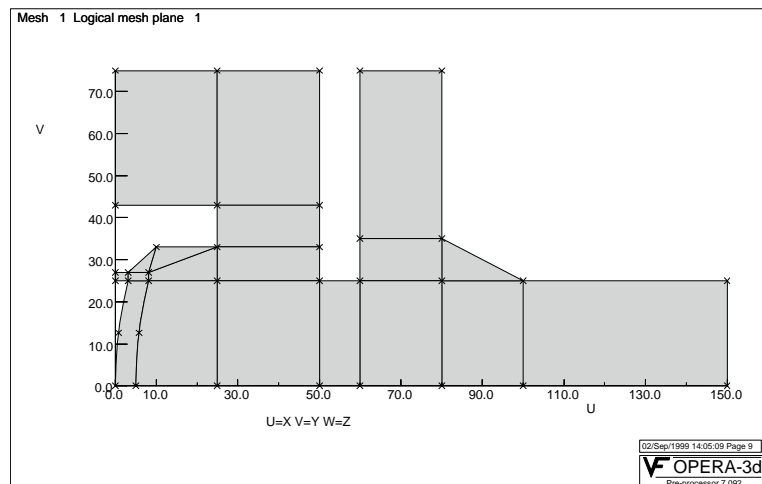


Figure 12.1 The longitudinal cross section of the gun. The electrons are emitted from the curved surface on the left, and travel toward the right under the influence of the fields created by the focusing electrodes.

Pre Processing, Constructing the Model

The Baseplane

We will first draw a 2-dimensional section which can then be extruded in a third direction to create the full 3-dimensional structure. In order to conserve time and computational resources it is always preferable, whenever possible, to exploit any symmetries present in a model. This particular geometry has a four-fold symmetry which we will take advantage of. The symmetry operations can always be restored later during post processing if desired.

Defining the Baseplane Points

We will choose the z direction as the extrusion direction and our baseplane will be the cross section shown in Figure 12.1. After launching the pre processor from the OPERA console window, select:

DEFINE ↓

Define new mesh → Finite element mesh → XY plane,
extrude in Z

and set:

W coordinate of plane = 0

We are then asked to specify an initial size for the display:

Minimum on horizontal axis	=	0
Maximum on horizontal axis	=	150
Minimum on vertical axis	=	0
Maximum on vertical axis	=	75
<input type="button" value="Accept"/>		<input type="button" value="Dismiss"/>

and **Accept**.

Select **Point Input**.

The points for the baseplane facets can be entered by specifying their coordinates using: **Give U, V, W**.

The list of coordinates that should be specified is:

(0, 0, 0),	(5, 0, 0),	(25, 0, 0),	(50, 0, 0),
(60, 0, 0),	(80, 0, 0),	(100, 0, 0),	(150, 0, 0),
(.797, 12.6, 0),	(5.797, 12.6, 0),	(0, 25, 0),	(3.17542, 25, 0),

(8.17542, 25, 0),	(25, 25, 0),	(50, 25, 0),	(60, 25, 0),
(80, 25, 0),	(100, 25, 0),	(150, 25, 0),	(0, 27, 0),
(3.17542, 27, 0),	(8.17542, 27, 0),	(10, 33, 0),	(25, 33, 0),
(50, 33, 0),	(60, 35, 0),	(80, 35, 0),	(0, 43, 0),
(25, 43, 0),	(50, 43, 0),	(0, 75, 0),	(25, 75, 0),
(50, 75, 0),	(60, 75, 0),	(80, 75, 0),	

The points can also be defined using **Construction lines**, but that will be left as an exercise for the interested reader. Close the **Point definition** menu with **Return**.

Defining the Baseplane Facets

The facets can now be drawn by clicking on **Facet Input** and drawing the facets shown in Figure 12.1. Note that the baseplane contains only 3- and 4-sided facets. The fastest way of defining the 4-sided facets is to use the option **auto-close after 4**. Remember to use the **F1** key to hide and restore the menus. Having defined all facets, close the **Facet Definition** menu with **Return**.

Defining the Baseplane Subdivisions

We now move to the subdivision definition menu by choosing **Subdivision**, and set the subdivision by selecting **Set subdivision**, then complete the parameter box as shown:

Subdivision		
<input type="checkbox"/> 1	<input type="checkbox"/> 2	<input type="checkbox"/> 3
<input type="checkbox"/> 4	<input type="checkbox"/> 5	<input type="checkbox"/> 6
<input type="checkbox"/> 7	<input type="checkbox"/> 8	<input type="checkbox"/> 9
Other	<input type="text" value="10"/>	
<input type="button" value="Accept"/>		

and **Accept**. By selecting **Apply to line** we can apply the subdivision by clicking just inside the edges at (150, 10) and (125, 0).

Similarly, **Set subdivision**

Subdivision		
<input type="checkbox"/> 1	<input type="checkbox"/> 2	<input type="checkbox"/> 3
<input type="checkbox"/> 4	<input type="checkbox"/> 5	<input type="checkbox"/> 6
<input type="checkbox"/> 7	<input checked="" type="checkbox"/> 8	<input type="checkbox"/> 9
Other	<input type="text"/>	
<input type="button" value="Accept"/>		

Accept followed by **Apply to line**, and click near the edges at (15, 0), (40, 0), (15, 35), (0, 60) and (60, 60).

Then **Set subdivision**

Subdivision		
<input type="checkbox"/> 1	<input type="checkbox"/> 2	<input type="checkbox"/> 3
<input type="checkbox"/> 4	<input type="checkbox"/> 5	<input checked="" type="checkbox"/> 6
<input type="checkbox"/> 7	<input type="checkbox"/> 8	<input type="checkbox"/> 9
Other	<input type="text"/>	
<input type="button" value="Accept"/>		

followed by **Accept**, and then **Apply to line**, and click just near the edges at (2.5, 0), (70, 0), (90, 0), (90, 30) and (10, 43). Then **Set subdivision**

Subdivision		
<input type="checkbox"/> 1	<input type="checkbox"/> 2	<input checked="" type="checkbox"/> 3
<input type="checkbox"/> 4	<input type="checkbox"/> 5	<input type="checkbox"/> 6
<input type="checkbox"/> 7	<input type="checkbox"/> 8	<input type="checkbox"/> 9
Other	<input type="text"/>	
<input type="button" value="Accept"/>		

Accept followed by **Apply to line**, and click near the edges at (55, 0), (25, 27), (60, 30), (5, 30) and (8, 30). Finally **Set subdivision**

Subdivision

<input type="checkbox"/> 1	<input type="checkbox"/> 2	<input type="checkbox"/> 3
<input checked="" type="checkbox"/> 4	<input type="checkbox"/> 5	<input type="checkbox"/> 6
<input type="checkbox"/> 7	<input type="checkbox"/> 8	<input type="checkbox"/> 9

Other

Accept followed by **Apply to line**, and click near the edges at (1.5, 25) and (1.5, 27). If the subdivisions have been set correctly, the baseplane should appear as shown in Figure 12.2.

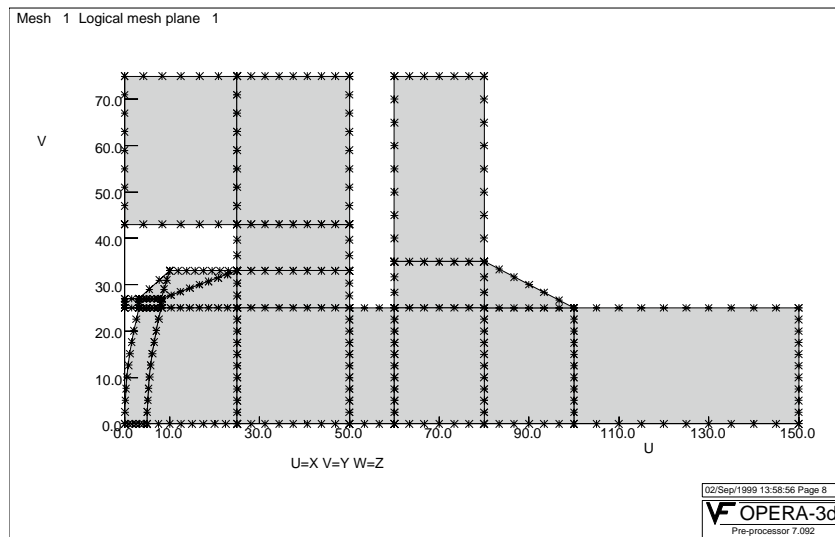


Figure 12.2 The subdivision definitions in the baseplane.

Close the **Subdivisions** menu by selecting the option **Check for Hex meshing**, which confirms that the model is suitable for hexahedral elements to be generated.

Creating the 3D Model

Defining the Extrusions

Now that the baseplane is complete, the three-dimensional model can be created. In this example the electrodes have a square cross section when viewed along the x-axis. The mesh will be created by rotating the base plane about the x-axis, and scaling the dimensions to create the required square cross section. This can be easily achieved by generating an extrusion of the baseplane onto the plane $Y = Z$, creating the first layer of the mesh which is a 45° wedge.

The first layer is created by selecting **Extrude** and then **Linear Extrusion** and completing the dialog box as shown:

Extrusion Definition	
Coordinate	<input type="text" value="0"/>
<input type="checkbox"/> Global	<input checked="" type="checkbox"/> Relative
Number of Elements	<input type="text" value="10"/>
<input type="button" value="Accept"/>	

and **Accept**.

This creates a second plane, coincident with the baseplane and generates the first layer of the mesh consisting of a set of regions that join the faces of the two planes.

The regions have zero volume, the points on the second plane need to be moved so that they lie on the plane $Y = Z$. Individual points could be selected now in order to redefine their positions, however if no points are selected any transformations will be applied to all points. Transform all the points by selecting **Transform Points**, then **Cartesian Coordinates** and completing the dialog box as follows:

Expression for U	=	U
Expression for V	=	V
Expression for W	=	V
<input type="button" value="Accept"/>		<input type="button" value="Dismiss"/>

Accept then **Return** to complete the transformations. The 45° wedge layer has now been created, select **Finish Editing** to move on to the material specification.

At this stage it is not necessary to define materials and boundary conditions. Select **Finish** from the materials menu, and from the next three menus for boundary conditions. This leaves the screen at the first level **Data Definition** menu.

A second extrusion can now be defined, select:

DEFINE ↓

Extend existing mesh → **Extend with editing**

Mesh number	=	1
Accept		Dismiss

After accepting, we select **ZX plane**, **extrude in Y** and then **Linear extrusion** and complete the dialog box as shown:

Extrusion Definition	
Coordinate	0
<input type="checkbox"/> Global	<input checked="" type="checkbox"/> Relative
Number of Elements	10
Accept	

and **Accept**. A third plane of points is created and a second extrusion layer with regions joining the faces on planes two and three. The points on the plane now need to be projected onto the $Y=0$ plane. As before select **Transform Points** → **Cartesian Coordinates**, and complete the dialog box:

Expression for U	=	U
Expression for V	=	V
Expression for W	=	0
Accept		Dismiss

and **Accept**. Select **Return** and **Finish Editing**. Then select **Finish** from the materials and boundary condition menus, followed by **Return** from the **Extend existing mesh** menu. Also close the **Data Definition** menu with **Return**.

Defining Materials and Potential Types

In this model all regions are air and should be assigned total scalar potential, and linear elements. This is the default setting, so no modifications are necessary.

Defining the Boundary Conditions

Since we are exploiting symmetry in this model we must apply appropriate boundary conditions at the symmetry planes. We also must specify the potentials on the focusing electrodes. Both symmetry planes (the xy and zx planes) require **TANGENTIAL ELECTRIC** boundary conditions. To do this we select

MODIFY ↓

Boundary conditions → Base plane → All external facets

and complete the dialog box:

Global Boundary Condition Definition		
Condition name:		
<input type="checkbox"/> Magnetic Scalar	<input type="checkbox"/> Normal Magnetic	<input type="checkbox"/> Tangential Magnetic
<input type="checkbox"/> Voltage	<input type="checkbox"/> Normal Electric	<input checked="" type="checkbox"/> Tangential Electric
<input type="checkbox"/> Total Ax	<input type="checkbox"/> Total Ay	<input type="checkbox"/> Total Az
<input type="checkbox"/> Incident Ax	<input type="checkbox"/> Incident Ay	<input type="checkbox"/> Incident Az
<input type="checkbox"/> Incident Voltage	<input type="checkbox"/> Perfect Conductor	<input type="checkbox"/> Radiation
<input type="checkbox"/> Normal Derivative	<input type="checkbox"/> Mixed Derivative	
<input type="checkbox"/> Symmetry	<input type="checkbox"/> Slip Surface	<input type="checkbox"/> Clear
Value	<input type="text"/>	Label/2nd value <input type="text"/>
<input type="button" value="Accept"/>	<input type="button" value="Help"/>	<input type="button" value="Quit"/>

followed by **Accept** and **Finish**. To set the voltages on the electrodes select

MODIFY ↓

Boundary conditions → Extrusion facets

Layer number	=	1
<input type="button" value="Accept"/>		<input type="button" value="Dismiss"/>

Accept, and then, to restore the display to the original view, select **Change view** and select **XY plane**, **extrude in Z** and **Accept**. Now choose **Select/de-select facet**, and then click on all the edges shown in Figure 12.3 that are marked by a heavy line (these lines show the intersection of the extrusion layer with the viewing plane).

Now choose **Select and define**, select the edge at (10, 43), and then complete the parameter box as shown followed by **Accept**.

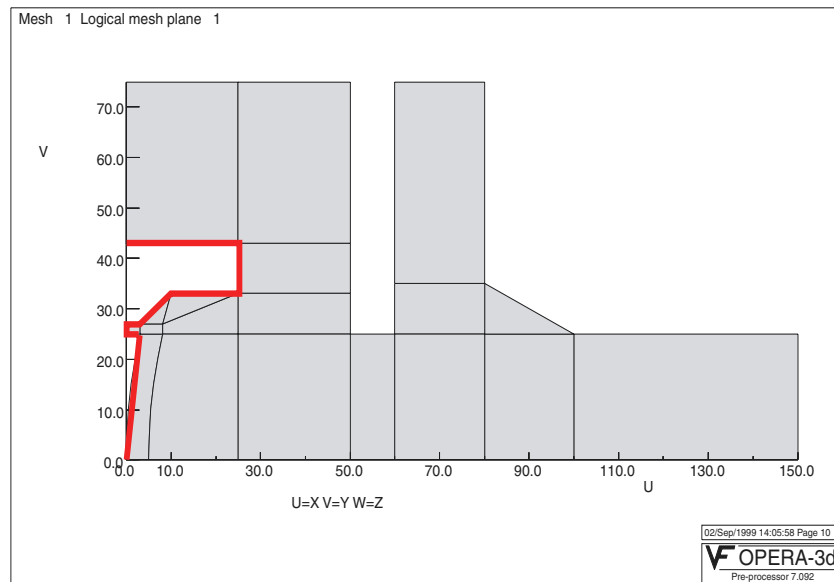


Figure 12.3 The extrusion facets which need to be assigned zero potential.

Boundary Conditions			
Condition name:			
<input type="checkbox"/> Magnetic Scalar	<input type="checkbox"/> Normal Magnetic	<input type="checkbox"/> Tangential Magnetic	
<input checked="" type="checkbox"/> Voltage	<input type="checkbox"/> Normal Electric	<input type="checkbox"/> Tangential Electric	
<input type="checkbox"/> Total Ax	<input type="checkbox"/> Total Ay	<input type="checkbox"/> Total Az	
<input type="checkbox"/> Incident Ax	<input type="checkbox"/> Incident Ay	<input type="checkbox"/> Incident Az	
<input type="checkbox"/> Incident Voltage	<input type="checkbox"/> Perfect Conductor	<input type="checkbox"/> Radiation	
<input type="checkbox"/> Normal Derivative	<input type="checkbox"/> Mixed Derivative		
<input type="checkbox"/> Symmetry	<input type="checkbox"/> Slip Surface	<input type="checkbox"/> Clear	
Value	<input type="text" value="0"/>	Label/2nd value	<input type="text"/>
Other volumes and layers:			
From	<input type="text" value="1"/>	To	<input type="text" value="*"/> <input type="checkbox"/> All facets
<input type="button" value="Accept"/>	<input type="button" value="Keep"/>	<input type="button" value="Help"/>	<input type="button" value="Quit"/>

Next choose **Select/de-select facet**, and then click on all the edges shown in Figure 12.4 that are marked by a heavy line.

Now choose **Select and define**, select the edge at (60, 60), and then complete the parameter box as shown followed by **Accept**.

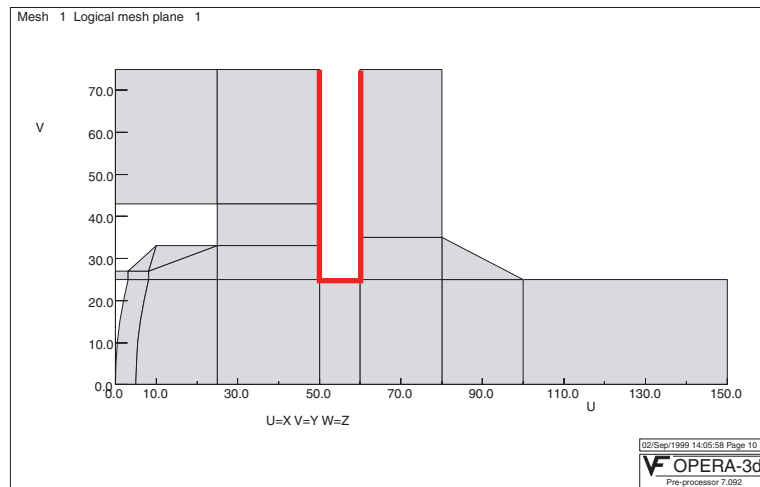


Figure 12.4 The extrusion facets which need to be assigned a potential of 10kV.

Boundary Conditions			
Condition name:			
<input type="checkbox"/> Magnetic Scalar	<input type="checkbox"/> Normal Magnetic	<input type="checkbox"/> Tangential Magnetic	
<input checked="" type="checkbox"/> Voltage	<input type="checkbox"/> Normal Electric	<input type="checkbox"/> Tangential Electric	
<input type="checkbox"/> Total Ax	<input type="checkbox"/> Total Ay	<input type="checkbox"/> Total Az	
<input type="checkbox"/> Incident Ax	<input type="checkbox"/> Incident Ay	<input type="checkbox"/> Incident Az	
<input type="checkbox"/> Incident Voltage	<input type="checkbox"/> Perfect Conductor	<input type="checkbox"/> Radiation	
<input type="checkbox"/> Normal Derivative	<input type="checkbox"/> Mixed Derivative		
<input type="checkbox"/> Symmetry	<input type="checkbox"/> Slip Surface	<input type="checkbox"/> Clear	
Value	<input type="text" value="10e3"/>	Label/2nd value	<input type="text"/>
Other volumes and layers:			
From	<input type="text" value="1"/>	To	<input type="text" value="*"/> <input type="checkbox"/> All facets
<input type="button" value="Accept"/>	<input type="button" value="Keep"/>	<input type="button" value="Help"/>	<input type="button" value="Quit"/>

Finally, choose **Select/de-select facet**, and then click on all the edges shown in Figure 12.5 that are marked by a heavy line

Then choose **Select and define**, select the edge at (125, 25), and then complete the parameter box as shown followed by **Accept**.

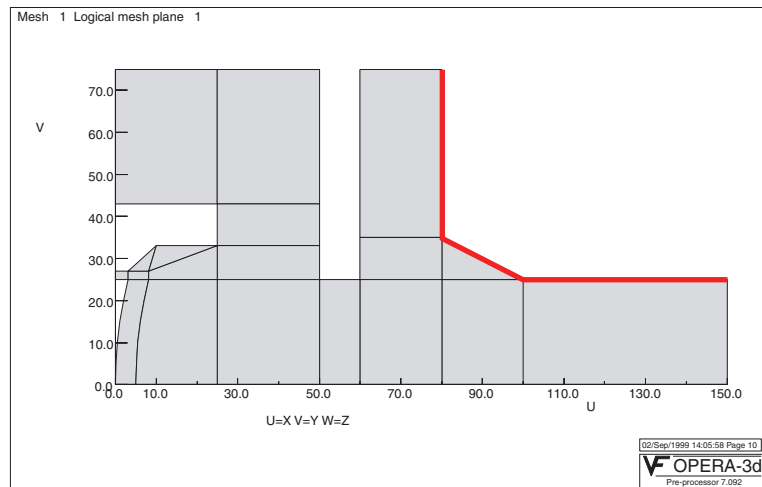


Figure 12.5 The extrusion facets which need to be assigned a potential of 20kV.

Boundary Conditions		
Condition name:		
<input type="checkbox"/> Magnetic Scalar	<input type="checkbox"/> Normal Magnetic	<input type="checkbox"/> Tangential Magnetic
<input checked="" type="checkbox"/> Voltage	<input type="checkbox"/> Normal Electric	<input type="checkbox"/> Tangential Electric
<input type="checkbox"/> Total Ax	<input type="checkbox"/> Total Ay	<input type="checkbox"/> Total Az
<input type="checkbox"/> Incident Ax	<input type="checkbox"/> Incident Ay	<input type="checkbox"/> Incident Az
<input type="checkbox"/> Incident Voltage	<input type="checkbox"/> Perfect Conductor	<input type="checkbox"/> Radiation
<input type="checkbox"/> Normal Derivative	<input type="checkbox"/> Mixed Derivative	
<input type="checkbox"/> Symmetry	<input type="checkbox"/> Slip Surface	<input type="checkbox"/> Clear
Value	<input type="text" value="20e3"/>	Label/2nd value <input type="text"/>
Other volumes and layers:		
From	<input type="text" value="1"/>	To <input type="text" value="*"/> <input type="checkbox"/> All facets
<input type="button" value="Accept"/>	<input type="button" value="Keep"/>	<input type="button" value="Help"/> <input type="button" value="Quit"/>

Creating the Mesh and Displaying the Model

We can now create the finite element mesh in preparation for creating an analysis database. We will create a hexahedral mesh by choosing

```
MESH ↓
  Surface mesh ...
    ...quadrilaterals
```

click on **Continue** to clear the message box, then pick on

MESH ↓
Volume mesh ...
Mesh

and hit **Continue** again.

To display a three-dimensional view of the model select

DISPLAY ↓
Display Command...
...view

and complete the parameter box:

Display View			
Size	<input type="text" value="75"/>		
Eye position:			
X	<input type="text" value="-1"/>	Y	<input type="text" value="1"/>
Z	<input type="text" value="-1"/>		
Centre of picture:			
X	<input type="text" value="50"/>	Y	<input type="text" value="50"/>
Z	<input type="text" value="0"/>		
Rotate picture	<input type="text" value="0"/>		
<input checked="" type="checkbox"/> New picture	<input type="checkbox"/> Add to picture		
<input checked="" type="checkbox"/> Parallel view	<input type="checkbox"/> Perspective view		
<input checked="" type="checkbox"/> Show Axes	<input type="checkbox"/> No axes		
<input type="button" value="Refresh Display"/>		<input type="button" value="Accept"/>	<input type="button" value="Quit"/>

followed by **Accept**. Then select

DISPLAY ↓

Display Command...

...style

Display Style	
Line view	<input type="checkbox"/>
Surface view	<input checked="" type="checkbox"/>
Full surface algorithm	<input type="checkbox"/>
No Elements	<input type="checkbox"/>
Surface Elements	<input checked="" type="checkbox"/>
Volume Elements	<input type="checkbox"/>
Vectors...	<input type="checkbox"/>
... no vectors	<input checked="" type="checkbox"/>
... in conductors only	<input type="checkbox"/>
... material orientation	<input type="checkbox"/>
... current density	<input type="checkbox"/>
... velocity	<input type="checkbox"/>
Refresh display	*
Return	←

and **Return**, followed by:

DISPLAY ↓

Display Command...

...select parts → Volume labels

and select **ALL** from the list followed by **Accept**, **Refresh display**, and then **Return** twice.

The resulting display is shown in Figure 12.6.

We can also check that the proper voltages have been assigned by selecting

DISPLAY ↓

Display Command...

...select parts → Facet labels

and choosing **Voltage** from the list, **Accept** and **Refresh display**, giving the display shown in Figure 12.7.

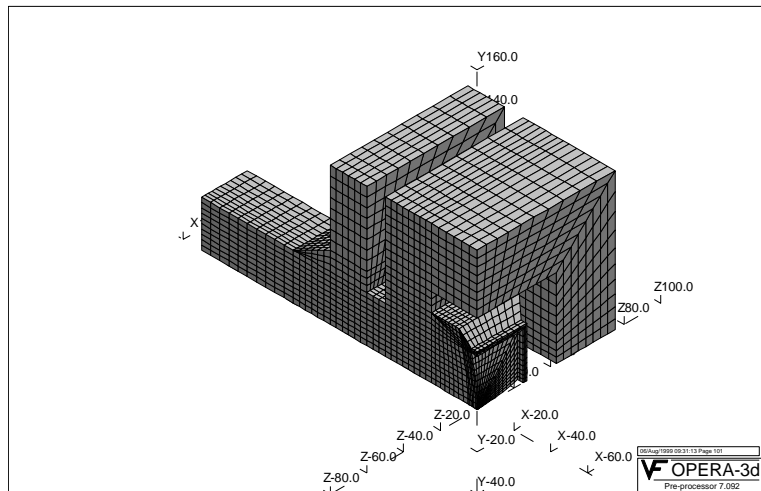


Figure 12.6 A 3D view of the model and finite element mesh.

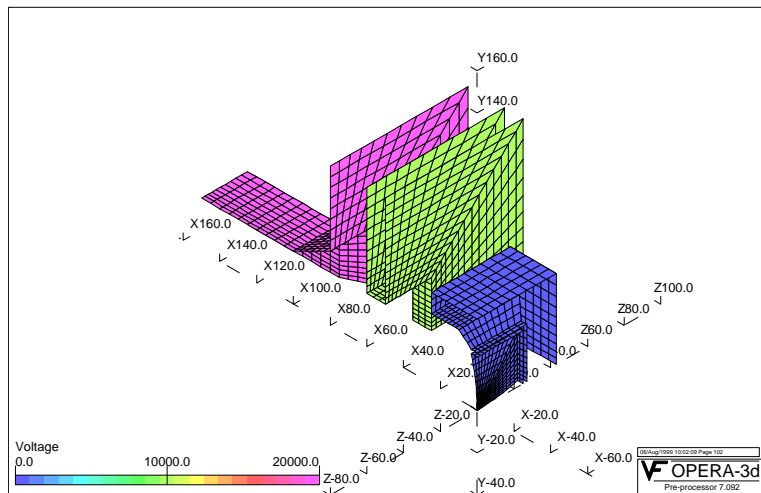


Figure 12.7 The assigned potentials on the electrode surfaces.

Creating the SCALA Database File

In order to perform a space charge analysis on this model we must first create a database file by selecting

FILE ↓

Create new database → Space charge beam (SCALA)

New Analysis Database File

File

Units: ☐ CGS ☒ SI (metres) ☒ SI (mm) ☐ SI (microns) ☐ Inches

Element type: ☒ Linear ☐ Quadratic ☐ Mixed

and **Accept**.

A box appears prompting us for a Problem Title. Any desired text can be entered followed by a single ***** on its own line.

Clear the message box by clicking **Continue**.

The SCALA data menu now appears

SCALA	
Materials	→
Periodicity conditions	→
External fields	→
Scala iteration data	→
Check data	*
Return	←

As all regions in the model are air, no **Materials** will have to be defined.

Parameters that control the SCALA calculation can be set. Select **SCALA Iteration Data** and set the under-relaxation factor to **0.2** (default is 1.0). This parameter often needs to be adjusted to obtain reliable convergence with SCALA. If the relaxation parameter is too large the current in the beam will oscillate. **Accept** and select **Return** to complete the database.

We now want to write out a pre processor file which will contain our model as well as the analysis options we have chosen. To do this select

FILE ↓

Write pre-processor file

File Selection Box	
Filter	
*.oppre	
Files in: current path	Subdirectories
	..
Selected file	
egun.oppre	
Selected directory	
Accept	Filter
CD	Dismiss

and **Accept**. Click on **Continue** to clear the message box and then **End OPERA-3d/Pre** to exit the pre processor.

Creating the Emitter File

The emitter file defines the sources of charged particles to be used in the model. The type of emission expected for each source must be defined and the geometry of the emission surface must be given. The file should have the same name as the database file created in the previous pages, but with a *.emit* extension.

The most reliable results will be obtained if the emitter surfaces exactly match the meshed surfaces of the model. The emitter file allows each emitter to be modelled as a set of surface panels: match these to the mesh, with the same subdivision used for the surfaces modelling the geometry.

This example uses a thermionic cathode as the emitter, solving Child's Law to calculate the emission current density. With this type of emission the **Normal sampling distance** is an important parameter. It is set to 0.4cm in this example. It should be small compared to the feature sizes and curvature of the emitter.

0	1	0	1			
1	0.2	0.001	0.4			
2	1975	4.0	120.0			
1.0	-1					
1	1	0.25				
2	0.01	0.0	0.0	0.0	0.0	0.0

```
2
10      10
0.0     0.0     0.0
.318    2.5     0
.318    2.5     2.5
0.0     0.0     0.0
0.08    1.26    0.0
0.318   2.5     1.25
0.08    1.26    1.26
0.0     0.0     0.0
2
10      10
0.0     0.0     0.0
.318    2.5     2.5
.318    0.0     2.5
0.0     0.0     0.0
0.08    1.26    1.26
0.318   1.25    2.5
0.08    0.0     1.26
0.0     0.0     0.0
```

The pre processor data file (*.oppre*) and the emitter file shown above (*.emit*) are provided with the OPERA installation (see the sub-folder *Examples/3D*).

Running the SCALA Simulation

UNIX Operating Systems

From within the OPERA-3d environment select the SCALA option

Option:

scala

and from the following prompt:—

Please give SCALA database filename (without the .OP3 suffix)

egun

and choose that the analysis is carried out immediately:

Do you want to run the analysis now or later? (n or l)

n

The analysis then proceeds automatically.

Windows Systems

Choose interactive solution under the OPERA-3d menu in the OPERA Console. Select **SCALA** from the list of solvers and then browse to where the *egun.op3* file was saved and select it. The analysis module will then proceed with the calculations.


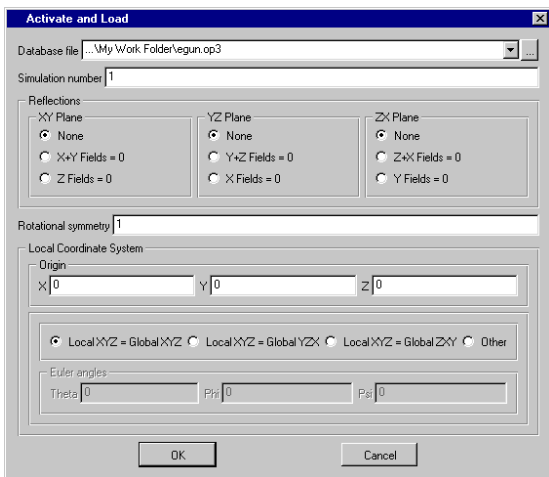
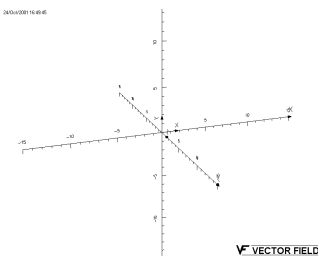

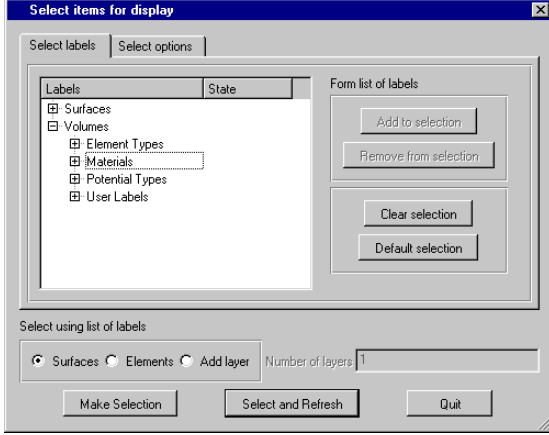
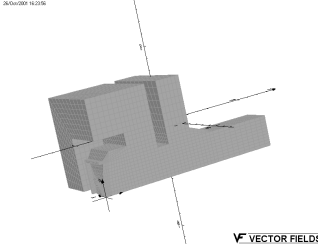
SCALA will run for 21 iterations, with the current converging to a current of 0.181 Amps (this is only the current in the section of the model that was meshed).

Close the **SCALA Analysis** window when the calculation has completed.

Post Processing, Analyzing the Results

Loading and Displaying the Model in the Post Processor

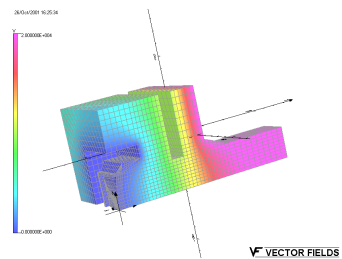
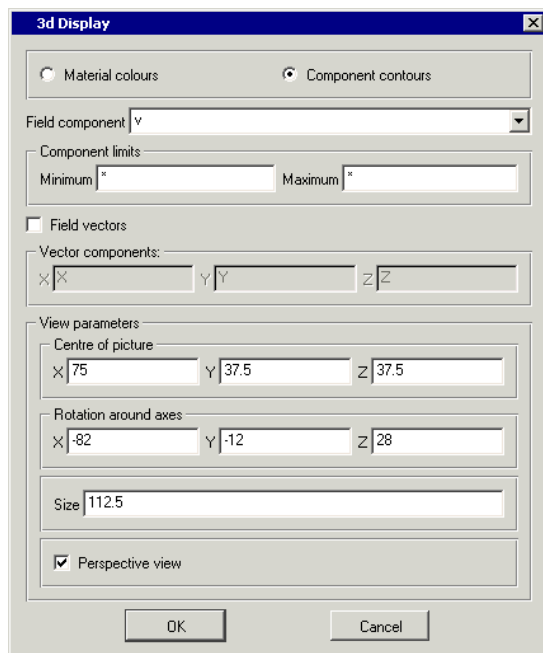
The solution file is read into the post processor and displayed. In order to see all of the materials, including the air regions, all of the materials are selected for display.

<p>Select the Open icon</p> 		
<p>Select the Select icon</p> 	<p>Click on the Materials tree entry and then use the Add to selection button to select all materials.</p>  <p>Pushing the Select and Refresh button selects the desired objects and updates the display.</p>	

Select the **3d display** icon


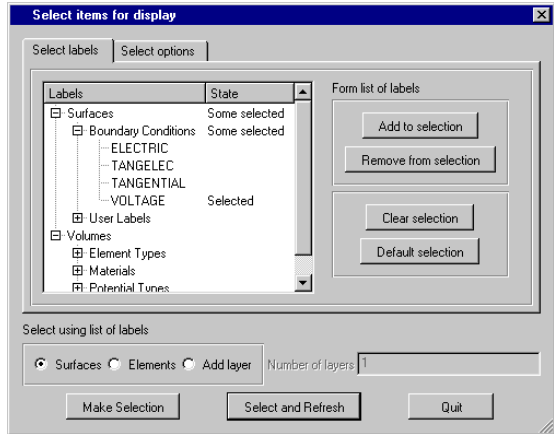
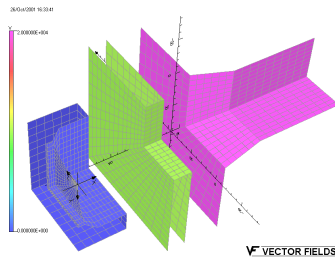
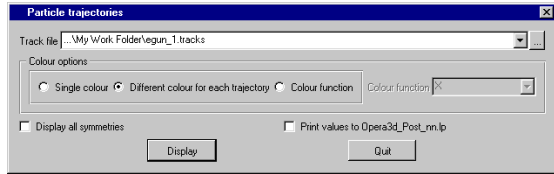
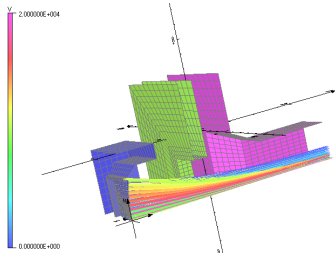


Now the surface of the model can be displayed showing the voltages.



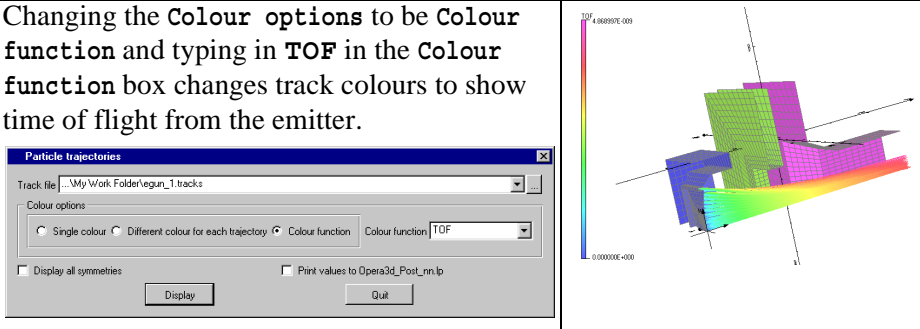
Displaying the Particle Tracks and Beam Cross Section

Now the model will be redisplayed showing only the VOLTAGE surfaces so that the particle trajectories can be seen more easily.

<p>Select the Select icon</p> 	<p>Click on the Materials tree entry and then use the Remove from selection button to de-select all materials. Then click on the VOLTAGE tree entry under Surfaces and use the Add to selection button to add all VOLTAGE boundaries to the display.</p>  <p>Pushing the Select and Refresh button selects the desired objects and updates the display.</p>	
<p>Trajectories → Display particle trajectories</p>		
	<p>From the Trajectories menu the TRACK file can be selected for display.</p> 	

Trajectories → Display particle trajectories

Changing the Colour options to be Colour function and typing in TOF in the Colour function box changes track colours to show time of flight from the emitter.



The screenshot shows the 'Particle trajectories' dialog box with the 'Colour function' option selected and 'TOF' entered in the 'Colour function' box. To the right is a 3D plot of particle trajectories, where the color of the lines represents the time of flight (TOF) from the emitter, as indicated by the color bar on the left of the plot.

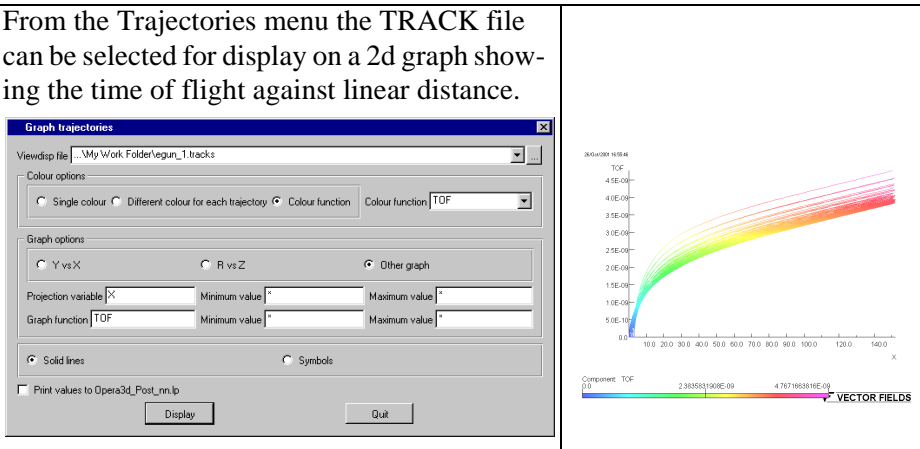
If the model is loaded in with all reflections active then the tracks display will be seen in all the reflections as well. In this model the reflections can be activated by adding in the reflections in the XY and ZX plane when opening the file.

Graph of Trajectories

There are other functions available for processing the particle trajectories. In addition to displaying the trajectories on the 3d model, they can also be plotted on 2d graphs.

Trajectories → Graph trajectories

From the Trajectories menu the TRACK file can be selected for display on a 2d graph showing the time of flight against linear distance.


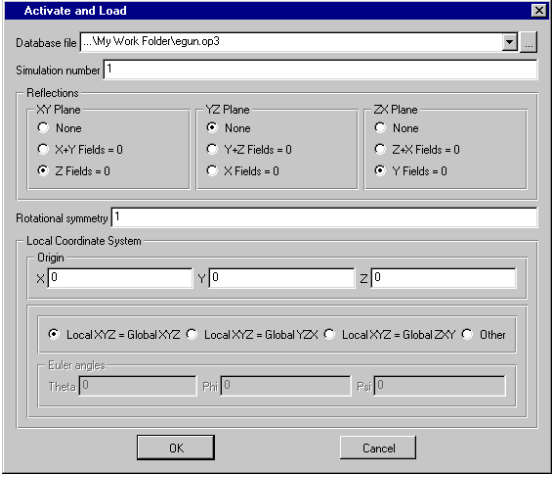


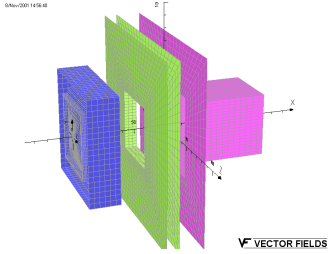


The screenshot shows the 'Graph trajectories' dialog box with the 'Other graph' option selected. The 'Projection variable' is set to 'X', and the 'Graph function' is set to 'TOF'. To the right is a 2D plot of particle trajectories, where the x-axis represents linear distance and the y-axis represents time of flight (TOF). The color of the lines represents the time of flight, as indicated by the color bar on the left of the plot.

Trajectory
Intersections

Intersections of the trajectories with a surface can also be plotted.

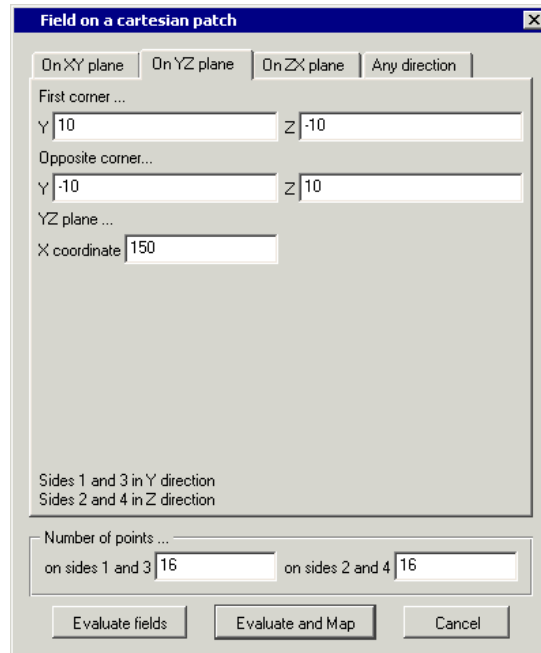
For this example define a 4 point cartesian patch at X=150, extending from Y = -10 to +10, Z = -10 to +10 with number of points set to 16 in both directions.

<div>Select the Open icon</div> <div></div>	<div>Firstly the complete model is loaded in with all reflections included.</div> <div></div>	
<div>Select the Default select and refresh icon</div> <div></div>	<div>The model can now be centred using the Initial view icon</div> <div></div>	<div></div>

Select the
**Fields on a
cartesian
patch** icon



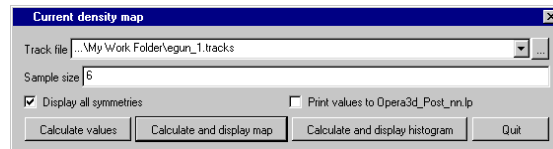
A surface for the intersections has to be defined first.



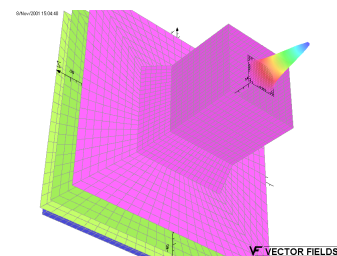
Pressing the **Evaluate fields** button creates the field map over the defined patch which is then used to find the intersections.

Trajectories → Current density map

Setting the FFT **Sample size** controls the smoothness of the plot. Selecting **Display all symmetries** show the intersections from all the tracks within the complete model.



Pressing the **Calculate and display histogram** button displays a histogram of the current intersecting the cartesian patch over the top of the model.



Chapter 13

A Cavity Example Using SOPRANO

Introduction

The models

Simple examples of a circular cavity and circular waveguide are described in this chapter. The model of the cavity consists of three layers. An entry port in layer 1 leads into the main cavity in layer 2, and there is an exit port in layer 3. The entry port has a radius of 50 mm, the main cavity has a radius of 250 mm, and the exit port has a radius of 50 mm again, see Figure 13.1.

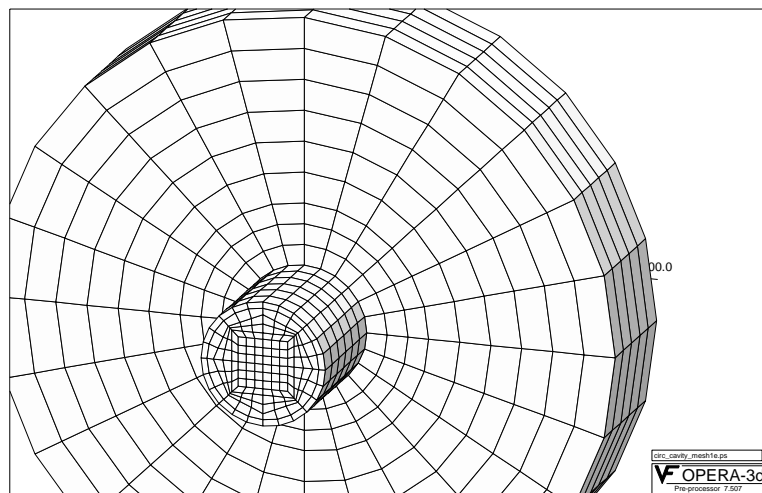


Figure 13.1 Mesh of the circular cavity in the pre processor

The thickness of the three layers is 100 mm each, so the model has a total length of 300 mm, see Figure 13.2.

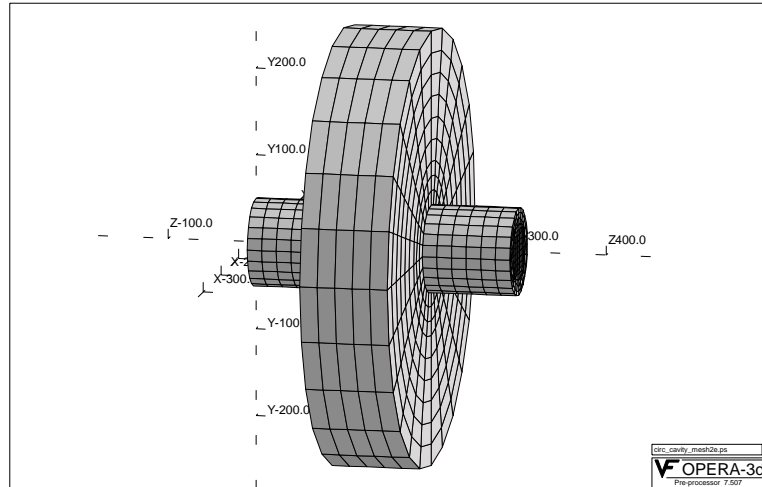


Figure 13.2 The cavity with entry port, main cavity and exit port

Only the inner volume of the cavity has been modelled with **AIR VECTOR**. The walls of the cavity have not been modelled with elements; on the outer walls the boundary condition **PEC** (perfect electric conductor) has been set. The model is excited with a functional potential boundary condition on the baseplane. At the exit port on the top plane a radiation boundary condition is applied to allow all energy to leave the problem. It should be noted that there is no air surrounding the cavity.

In order to save computer resources in terms of elements and equations, the two rings in layer 1 and in layer 3 have been “nulled out”. Figure 13.3 shows the material properties in layer 1. Volumes with the label **NULL** are omitted from the final mesh.

Although the cavity has got a rotational symmetry, all 360° have been modelled. This allows the calculation of all eigenvalues without further thinking about modes and boundary conditions.

The mesh of the cavity intentionally has been defined quite coarsely in order to speed up the calculations and obtain a first solution. The mesh should be refined for more accurate results, and particularly for the calculation of higher modes.

Figure 13.4 shows the facets on the baseplane which have actually been drawn to define the baseplane. All other facets have been created with the copy/mirror

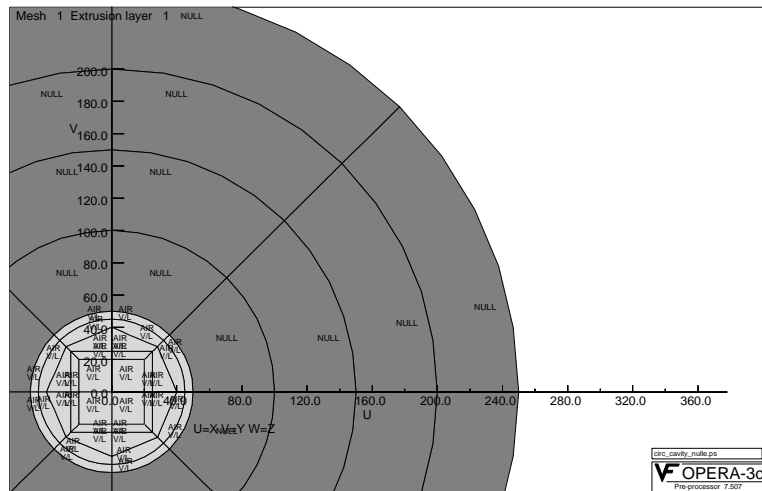


Figure 13.3 The outer ring in layer 1 has been “nulled out”

commands in group operations. It is important that in OPERA-3d a facet should have a maximum angle of 45° in order to model a circle with good accuracy.

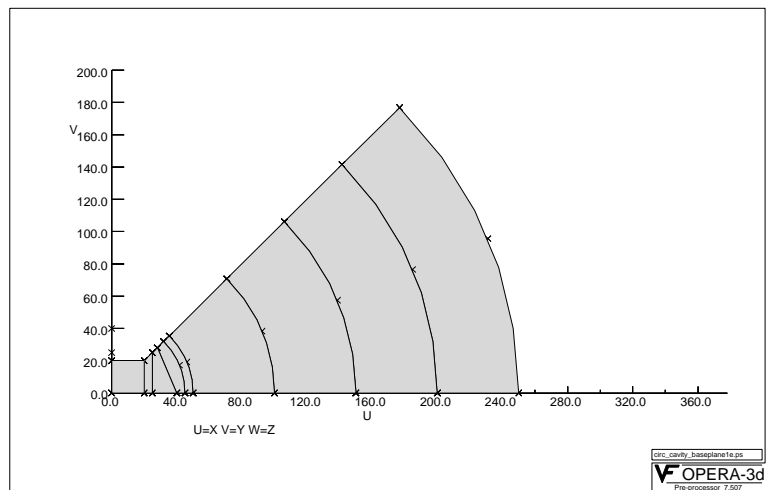


Figure 13.4 Facets on the baseplane needed to set up the cavity

Along the central z-axis of the cavity there are square elements to ensure a regular mesh. This is also important to achieve a good accuracy in the region of most interest.

Running the Eigenvalue Solver

In a first step the eigenvalues (or eigenfrequencies) are calculated. The lowest eigenfrequency is often of most interest. In a second step the circular guide feeding the cavity can be analysed with the steady-state solver.

The pre processor data file (*.oppre*) is provided within the OPERA installation (see the sub-folder *Examples/3D*). Any functional potential boundary conditions or radiation boundary conditions are ignored for the SOPRANO-EV analysis. These conditions are not relevant to the eigenvalue solver, and so the program will apply the natural boundary condition on these surfaces, which is **TANGENTIAL ELECTRIC**. All other surfaces of the model have the **PEC** condition, which is equivalent to **NORMAL ELECTRIC**.

In literature we can find an equation for the first (lowest) eigenfrequency of a circular cavity:

$$f = \frac{2.405 \cdot c}{2 \cdot \pi \cdot r} = \frac{2.405 \cdot 3 \times 10^8}{2 \cdot \pi \cdot 0.25} = 4.59 \times 10^8 \quad . \quad (13.1)$$

Expecting the first (lowest) eigenfrequency in the region of 500 MHz, the SOPRANO-EV solver can be set up.

To run the analysis, use the data (*.oppre*) file provided. Create the surface (quadrilateral) and the volume meshes, and then create a new database. Set the number of eigenvalues to search for to be 10, and the range to be 500 to 1000 MHz. The analysis can be run from the pre processor.

The program searches for eigenfrequencies starting from the centre of a given range. However, the eigenfrequencies found are not strictly confined to the range. For example the cavity discussed here gives the following results:

```
Requesting Frequencies in Range [500.00      ,1000.0      ] (Mhz)
Opening database circ_cavity_ev1.op3
Eigen-Frequency  1 =  468.29      (Mhz)
Eigen-Frequency  2 =  728.43      (Mhz)
Eigen-Frequency  3 =  994.70      (Mhz)
Eigen-Frequency  4 =  994.72      (Mhz)
Eigen-Frequency  5 = 1093.6       (Mhz)
```

The solver can only calculate a limited number of eigenfrequencies in one calculation. In this case it has found 5 eigenfrequencies. It also has reported the first eigenfrequency of 468.29 MHz, which is near to the expected value, but actually slightly outside the given range. The higher frequency computed by SOPRANO-EV results from the entry and exit pipes in the circular cavity.

Run the analysis again, this time with a frequency range 1000 MHz to 1500 MHz. The following eigenvalues will be found:

```
Requesting Frequencies in Range [1000.0      ,1500.0      ] (Mhz)
Opening database circ_cavity_ev2.op3
Eigen-Frequency  1 =   1093.6      (Mhz)
Eigen-Frequency  2 =   1249.0      (Mhz)
Eigen-Frequency  3 =   1249.0      (Mhz)
Eigen-Frequency  4 =   1335.1      (Mhz)
Eigen-Frequency  5 =   1430.8      (Mhz)
```

It can be seen that one eigenfrequency is found in both ranges. This makes sure that no frequency has been omitted. Figure 13.5 shows the magnitude of the electric field distribution for the 9 eigenfrequencies listed above. Some of the eigenfrequencies are essentially duplicated, with the same mode excited but rotated in the model by a half period. This results from the discretisation of the circular cavity.

Losses in the wall

Although the wall of the cavity has not been modelled with elements, it is possible to calculate the losses in the wall. The following *.comi* file helps to define the parameters which are necessary (this is also available in the *installation* subfolder *Examples/3D*). After having run the *.comi* file, the losses in the wall can be displayed with **COMP = #LOSS**. Figure 13.6 then shows the losses on the surface.

```
/===== Begin of LOSS.COMI =====
/ Set SI units
unit len=metr,flux=tesl,fiel=am,elec=vm,s=amp,v=wbm,
co=sm,cu=am2,p=watt,fo=newt,ener=joul
/
/ Select the PEC (metallic) surfaces
select action=remove label=all_surfaces
select action=add label=pec
select action=select option=surfaces cut=no
/
/ Define variables for abs(Hxn)**2, omega,
/ conductivity (for copper say 5e7 S/m), etc.
/
$para #hxn (nz*hy-ny*hz)**2+(nx*hz-nz*hx)**2+(ny*hx-nx*hy)**2
$para #omeg 2*pi*freq
$cons #cond 5e7
$para #fact sqrt(#omeg*mu0/2/#cond)
$para #loss #fact*#hxn
/===== End of LOSS.COMI =====
```

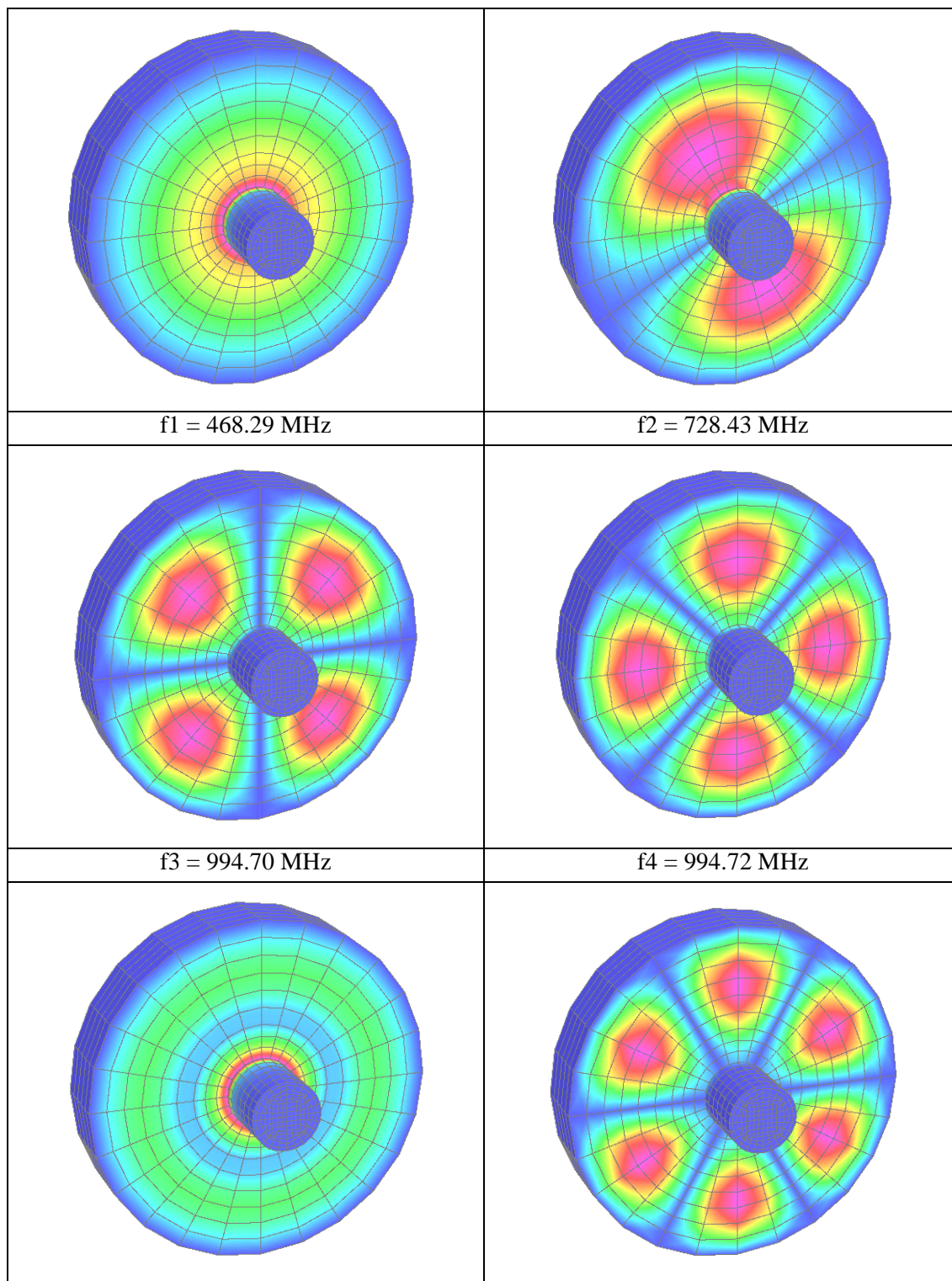


Figure 13.5 The 9 eigenfrequencies / modes found in the cavity

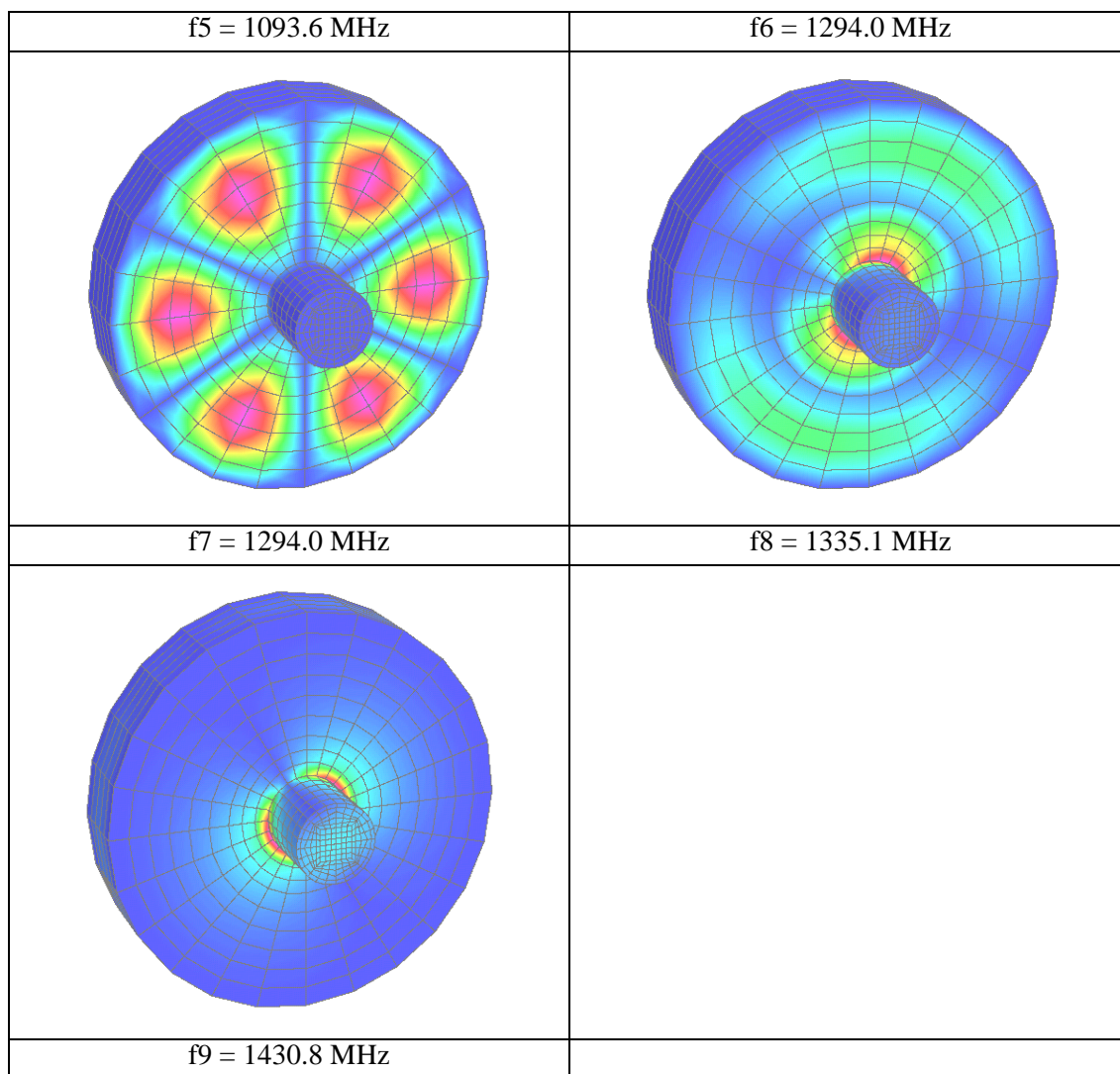


Figure 13.5 The 9 eigenfrequencies / modes found in the cavity (*continued*)

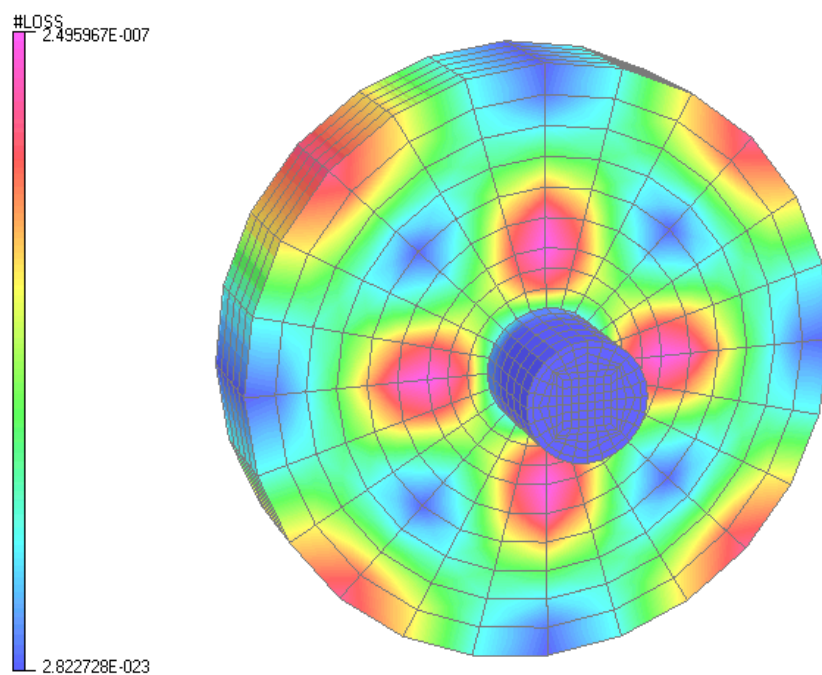


Figure 13.6 Power losses in the wall of the cavity for $f_3 = 995$ MHz

Running the Waveguide Analysis

Rather than SOPRANO-EV, which can find solutions only for the resonant frequencies, SOPRANO-SS calculates the field for any given frequency. However the model has to be excited with a particular driving function.

The circular cavity model is modified to produce a circular waveguide by making the regions of the cavity model at radius greater than 50 mm as a **NULL** material. A **PEC** boundary condition is applied to all the boundary at 50 mm. The problem is driven by a circular TM_{01} mode defined by the vector potential

```
AX X*COS(#R/44) keep
AY Y*COS(#R/44)
```

on the input port on the baseplane.

The model can be found in the installation sub-folder *Examples/3D*. The parameter **#R** has been previously defined in the first lines of the *.oppre* file before the **DEFINE** command starts:

```
$PARAM #R SQRT(X*X+Y*Y)
```

Create the surface (quadrilateral) and volume mesh, and create a new database. The cutoff frequency for this mode in the guide is about 2.3 GHz, so the problem is excited at 2.5 GHz to ensure propagation. Run the analysis from the pre processor. View the results in the post processor. As an example, Figure 13.7 shows the real part of the X-component of the electric field at this frequency.

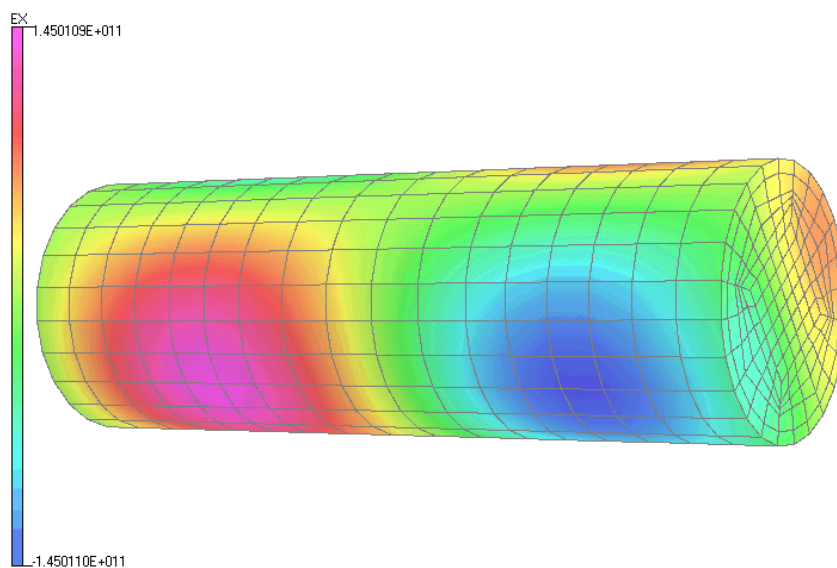


Figure 13.7 Circular waveguide excited at 2.5 GHz.

The propagation in the guide and the effect of the **RADIATION** boundary condition can be seen by examining the electric field on a line along the guide. Figure 13.8 shows the real and imaginary parts of the components of the electric field along the line (20,10,0) to (20,10,300). Note the use of the system variables **REX**, **IEY** etc. to denote the real and imaginary parts.

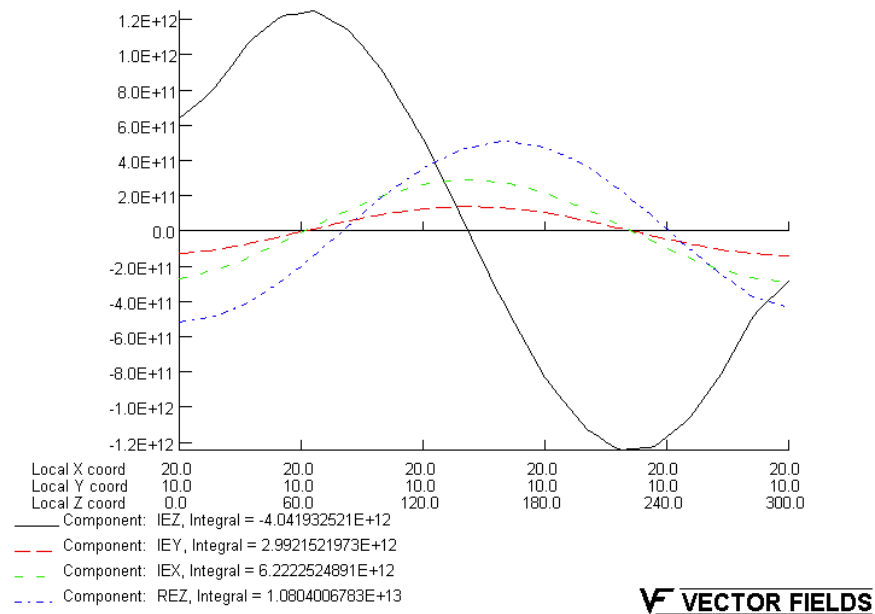


Figure 13.8 Electric field along waveguide

Discussion

For higher frequencies and shorter wavelengths the discretisation of the models needs to be refined. The effects of this can be seen in the steady state solution where the real and imaginary parts do not have the same peak values. At some point the limits of the computer system are reached in terms of nodes and elements. To overcome this, only half or only a quarter of the cavity or guide can be modelled.

On the resulting symmetry boundaries appropriate boundary conditions have to be applied to allow the modes which have been found in the eigenvalue solution. A printout of the first modes as seen in Figure 13.5 can help to choose the correct boundary condition. For higher modes the FE-mesh can be reduced even further to cut down the size of the model.

INDEX

Symbols

\$vfdir 9-36

-A-

accuracy 6-6, 6-11, 6-16, 6-28, 6-45
 CARMEN 6-43
 ELEKTRA 6-28
 SOPRANO 6-53
 TOSCA 6-13
 analysis
 ELEKTRA 8-52
 TOSCA 8-41, 9-1
 anisotropy 3-3, 6-26, 6-41
 automatic mesh generation 3-5

-B-

baseplane 8-4, 9-3
 BH curves 3-3, 9-35
 BH files
 Unix 2-5
 Windows 2-18
 boundary conditions ... 6-2, 6-4, 6-12, 6-26,
 6-41, 6-52, 8-23, 9-27
 current flow 6-12
 default 9-27
 CARMEN 6-42
 ELEKTRA 6-27
 SOPRANO 6-53
 TOSCA 6-13
 electrostatics 6-12

essential 6-13, 6-28, 6-43
 functional 3-6
 mixed 6-12
 open boundaries .. 6-14, 6-18, 6-29, 6-33,
 6-44, 6-48, 6-53
 periodicity 6-15

-C-

CHECK command 3-5
 Child's law 12-16
 COMI Files 2-5, 2-17
 command files 2-5, 2-17
 conductor field 6-10, 6-20, 6-36
 conductors 6-55, 8-29, 9-30
 construction lines 8-5, 8-6, 8-7, 9-4
 continuity 6-4, 6-11
 Coulomb gauge 6-21, 6-37
 CPU priority
 Windows 2-11

-D-

data files
 pre processor 8-39, 8-57, 9-38
 database
 ELEKTRA 8-53
 TOSCA 8-36
 display 8-26, 8-44, 9-41
 dongle installation 2-9
 drawing regions 5-33
 drive label 6-25, 6-40

-E-

electric scalar potential 6-21, 6-37
 electron gun 11-1
 ELEKTRA 8-58
 emitter data 6-64
 emitter files 11-2, 11-6, 12-16
 emitters 6-55, 11-2, 11-6
 environment variables
 Unix 2-2
 extrusions 8-15, 9-13
 global 9-14
 local 9-14

-F-

facets 8-11, 9-8
 field emission 6-63
 file names
 Unix 2-4
 Windows 2-16
 finite element method 6-2
 force 9-45
 Fowler Nordheim field 6-62

-G-

Galerkin 6-3
 graphics variables
 Unix 2-2
 Green's theorem 6-4

-I-

integral fields 3-9

-L-

least squares 6-3
 licenses 2-10
 log files
 Unix 2-4
 low frequency limit 6-20, 6-36

-M-

magnetic field 6-55

material orientation 3-3, 9-24
 material properties 3-3, 8-17, 8-54, 9-17,
 9-18
modeller.comi 2-5, 2-18
 modifying data 9-25
 motion 6-25
 multiple drives 3-2
 multiply connected 9-21

-N-

Newton-Raphson method 6-5
 nodally averaged fields 3-9
 non-linear materials 3-3, 6-26, 6-41

-O-

OpenGL Libraries 2-1
 OPERA Console 2-10
opera.comi 2-5, 2-17
 OPERA-2d 5-33
 OPERA-3d 3-1, 6-19, 6-35, 6-49
 OPERA-3d philosophy 3-2
Opera3d_Modeller_n.log
 Unix 2-4
 Windows 2-17
Opera3d_Modeller_n.lp
 Unix 2-4
 Windows 2-17
Opera3d_Post_n.log
 Unix 2-5
 Windows 2-17
Opera3d_Post_n.lp
 Unix 2-5
 Windows 2-17
Opera3d_Pre_n.backup
 Unix 2-5
 Windows 2-17
Opera3d_Pre_n.log
 Unix 2-5
 Windows 2-17
Opera3d_Pre_n.lp
 Unix 2-4

Windows 2-17
oppre.comi 2-5, 2-17

-P-

Peclet number 6-33
 periodicity 3-5
 permanent magnet 6-26, 6-41, 9-1, 9-24,
 9-35
 plasma free surface 6-64
 Point definition 9-4
 points 8-4, 8-7, 9-4
 cartesian coordinates 8-8, 8-9
 construction line intersections ... 8-7, 9-7
 cylindrical polar coordinates 8-9
 Poisson equation 6-2
 polygons 3-5
 post processing 3-8, 8-42, 9-40
 potential interface 6-11
 potential types 6-11, 6-15, 6-30, 6-45
 potentials 8-17, 8-18, 8-20, 8-53, 9-18
 pre processor 3-4
 program size
 Unix 2-6

-R-

reduced scalar potential 6-10, 6-20, 6-36
 reflections 9-45
 relaxation factor 6-5
 residual 6-5
 restarts 3-3
 Richardson-Dushman 6-56
 rotation 6-39
 Rutherford Appleton Laboratory 6-9

-S-

Schottky field emission 6-63
 shape function 6-3, 6-10, 6-27
 simulations 3-2, 8-53
SLIP command 6-38
 source currents 3-3, 6-11
 Space charge 11-1

Space charge analysis module 11-2, 11-7
 steady-state ac 6-22, 6-51
 stiffness matrix 6-4
 subdivisions 8-14, 8-15, 8-16, 9-11
 variable 9-28
 surface mesh 3-5
 swap space
 Unix 2-6
 system icon tray 2-10
 system manager 8-1
 Unix 2-3
 system variables 3-8

-T-

text window
 Windows 2-14
 thermal saturation 6-56
 thermionic cathode 12-16
 three dimensional model 6-6
 time harmonic 6-22, 6-51
 time tables 6-25, 6-40
 total scalar potential .6-11, 6-16, 6-31, 6-46
 transient analysis 6-23
 transient driving functions 6-24, 6-40
 two dimensional model 6-6

-U-

Unix implementation 2-1
 updating licenses 2-10
 upwinding 6-34

-V-

variational methods 6-3
 vector potential 6-20, 6-36
 velocity 3-3
 VFDIR
 see \$vfdir 9-36
 vfdir
 Unix 2-2
VFGRAHICS
 Unix 2-2

VFINV

Unix 2-3

VFWINDOWH

Unix 2-2

VFWINDOWW

Unix 2-2

virtual memory

Unix 2-6

voltage 3-8

voltage drive 6-29

volume mesh 3-5

-W-

weighted residuals 6-3

-X-

XY input 5-33